

MANUAL TECNICO

Propuesta de una aplicación web para el control de fondos y donaciones de
la ONG Oikos de Nicaragua

2016

**UNIVERSIDAD NACIONAL DE INGENIERÍA
FACULTAD DE ELECTROTECNIA
Y COMPUTACIÓN
UNI-FEC**



AUTORES:

**Br. Lester Eliezer Huembes López
Br. Juan Carlos Alemán Mayorga.**

TUTOR:

Msc.Ing. Jorge Prado.

**Managua, Nicaragua
Enero 2016**

Contenido

1. INTRODUCCIÓN.....	3
1. Interfaz de usuario.....	4
2. Menú del sistema.....	5
3. ESTRUCTURA DE DATOS	6
4. Clases principales del sistema	7

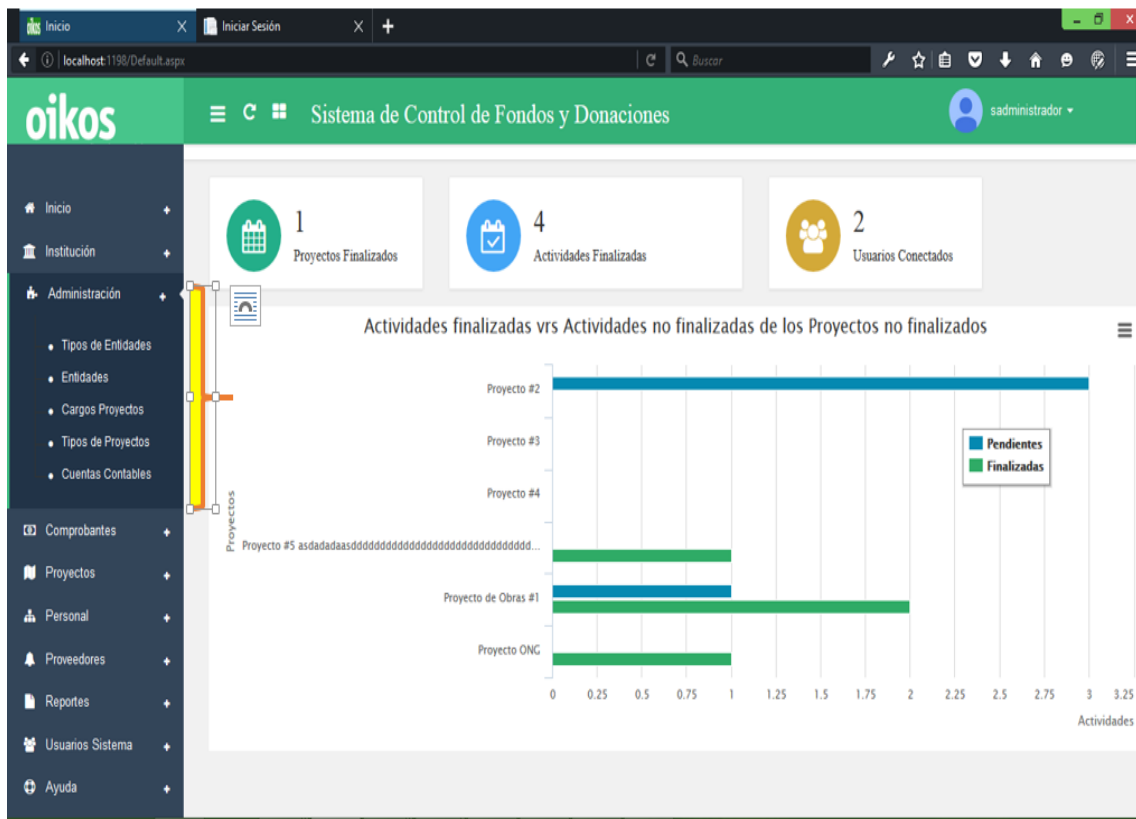
1. INTRODUCCIÓN

El siguiente documento describe aspectos técnicos del Sistema de Información web para control de fondos y donaciones de ONG OKOS. Dicha aplicación permitirá llevar a cabo el control de las donaciones y como se va utilizado el dinero destinado para un proyecto.

A continuación se detallan generalidades del diseño del sistema, estructura de la base de datos y explicación de códigos fuentes principales dentro de las clases. Los comentarios dentro del código fuente de las clases, explican el uso de cada segmento de código y funciones específicas.

1. Interfaz de usuario.

El diseño de la interfaz fue construido utilizando HTML controles ASP.NET y usando el framework Bootstrap. Se crea una página maestra que contiene el menú, y desde ahí se llaman a todos los formularios web que ocupan que son ocupados para realizar los procesos.

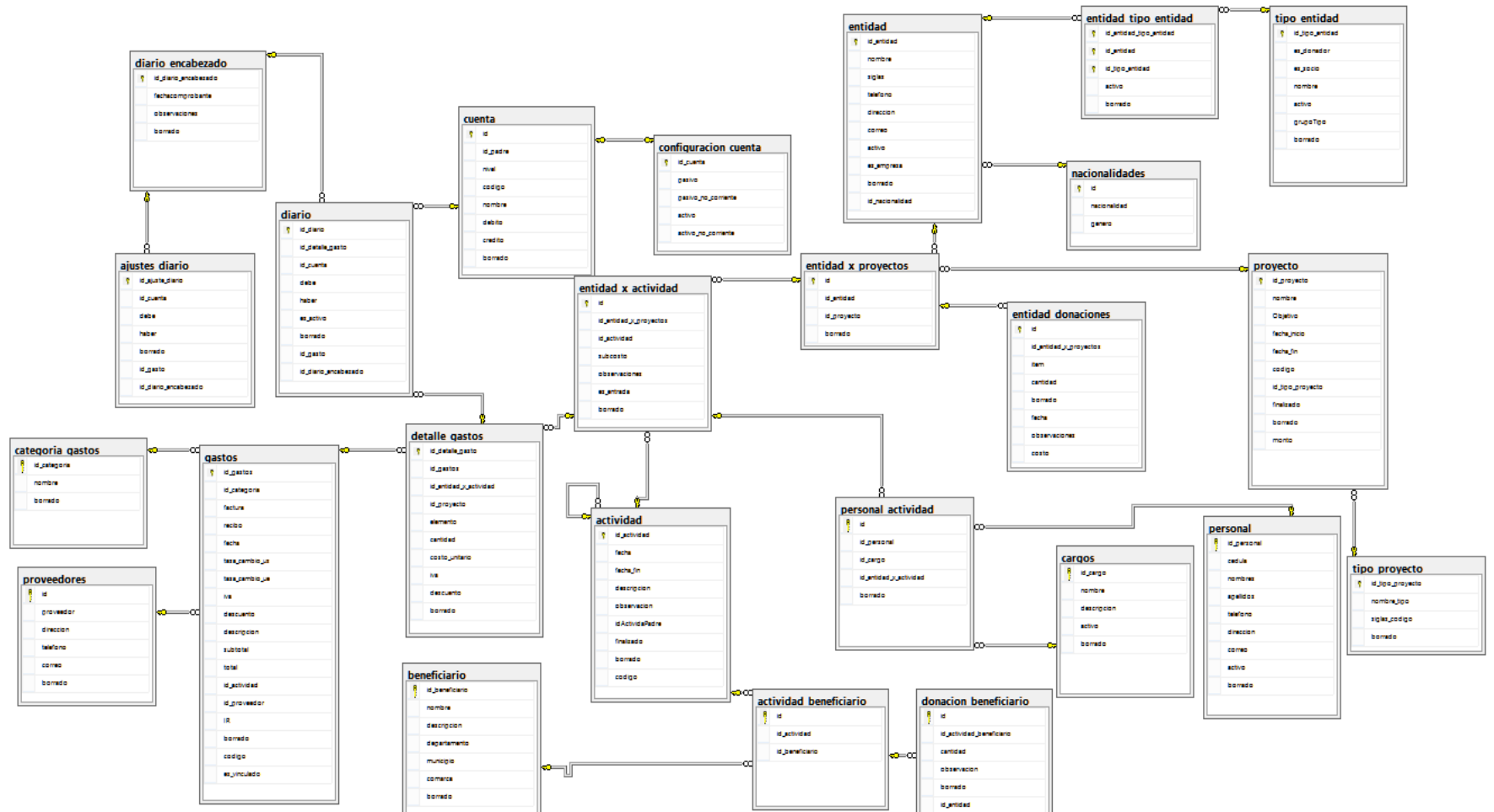


Cada elemento del menú redirecciona a una opción dentro del sistema; Tomando en cuenta el rol de los usuarios, debido a que hay niveles de permiso a la información que proporciona el sistema, los principales son el administrador, el consultor (el que solo puede ver los reportes) y el operador quien alimenta el sistema con información.

2. Menú del sistema

MENU	ELEMENTOS	PERFILES ACCESO
Administración	<ul style="list-style-type: none"> • Entidades • Tipos de proyecto • Catalogo cuentas contables • Cargos de proyecto 	Operador, Adminsitrador
Gestión de proyectos	<ul style="list-style-type: none"> • Proyectos • Actividades • Donaciones recibidas • Personal en actividad • Gastos de proyecto 	Operador, Adminsitrador
Beneficiario	<ul style="list-style-type: none"> • Catálogo de beneficiario por actividad y proyecto 	Consultor, Operador, Adminsitrador
Comprobante contable	<ul style="list-style-type: none"> • Comprobantes de pagos por gastos es decir por facturas. 	Operador, Adminsitrador
Gestión de usuario	<ul style="list-style-type: none"> • Agregar usuario • Eliminar usuario • Administrar contraseña 	Administrador
Reportes	<ul style="list-style-type: none"> • Reporte de actividad • Reporte de gasto • Reporte de proyecto • Reporte de comprobante • Reporte de personal • Reporte de Beneficiario • Reporte consolidado de proyectos 	Consultor, adminsitrador, Operador

3. ESTRUCTURA DE DATOS



4. Clases principales del sistema

```
01 namespace Entidadees
02 {
03     using System;
04     using System.Collections.Generic;
05
06     public partial class actividad
07     {
08         public actividad()
09         {
10             this.actividad1 = new HashSet<actividad>();
11             this.actividad_beneficiario = new HashSet<actividad_beneficiario>();
12             this.archivos_actividades = new HashSet<archivos_actividades>();
13             this.entidad_x_actividad = new HashSet<entidad_x_actividad>();
14         }
15
16         public int id_actividad { get; set; }
17         public Nullable<System.DateTime> fecha { get; set; }
18         public Nullable<System.DateTime> fecha_fin { get; set; }
19         public string descripcion { get; set; }
20         public string observacion { get; set; }
21         public Nullable<int> idActividaPadre { get; set; }
22         public bool finalizado { get; set; }
23         public bool borrado { get; set; }
24         public string codigo { get; set; }
25
26         public virtual ICollection<actividad> actividad1 { get; set; }
27         public virtual actividad actividad2 { get; set; }
28         public virtual
```



```

ICollection<actividad_beneficiario> actividad_beneficiario { get; set; }
29     public virtual ICollection<archivos_actividades> archivos_actividades { get; set; }
30     public virtual ICollection<entidad_x_actividad> entidad_x_actividad { get; set; }
31 }
32 }

```

```

01 namespace Entidadees
02 {
03     using System;
04     using System.Collections.Generic;
05
06     public partial class actividad_beneficiario
07     {
08         public actividad_beneficiario()
09         {
10             this.donacion_beneficiario = new HashSet<donacion_beneficiario>();
11         }
12
13         public int id { get; set; }
14         public int id_actividad { get; set; }
15         public int id_beneficiario { get; set; }
16
17         public virtual beneficiario beneficiario { get; set; }
18         public virtual actividad actividad { get; set; }
19         public virtual ICollection<donacion_beneficiario> donacion_beneficiario { get; set; }
20     }
21 }

```

```

01 namespace Entidadees
02 {
03     using System;
04     using System.Collections.Generic;
05
06     public partial class ajustes_diario
07     {
08         public int id_ajuste_diario { get; set; }
09         public int id_cuenta { get; set; }
10         public decimal debe { get; set; }
11         public decimal haber { get; set; }
12         public bool borrado { get; set; }
13         public int id_gasto { get; set; }
14         public int id_diario_encabezado { get; set; }
15
16         public virtual diario_encabezado diario_encabezado { get; set; }
17     }
18 }

```

```

01 namespace Entidadees
02 {
03     using System;
04     using System.Collections.Generic;
05
06     public partial class archivos_actividades
07     {
08         public archivos_actividades()
09         {
10             this.archivos_actividades1 = new HashSet<archivos_actividades>();
11         }
12
13         public int id { get; set; }
14         public string ruta_archivo { get; set; }
15         public int id_tipo_archivo { get; set; }
16         public Nullable<int> id_actividad { get; set; }
17         public Nullable<int> idPadre { get; set; }
18
19         public virtual ICollection<archivos_actividades> archivos_actividades1 { get; set; }
20         public virtual archivos_actividades archivos_actividades2 { get; set; }
21         public virtual tipo_archivos tipo_archivos { get; set; }
22         public virtual actividad actividad { get; set; }
23     }
24 }

```

```

01 namespace Entidadees

```

```

02 {
03     using System;
04     using System.Collections.Generic;
05
06     public partial class beneficiario
07     {
08         public beneficiario()
09         {
10             this.actividad_beneficiario = new HashSet<actividad_beneficiario>();
11         }
12
13         public int id_beneficiario { get; set; }
14         public string nombre { get; set; }
15         public string descripcion { get; set; }
16         public string departamento { get; set; }
17         public string municipio { get; set; }
18         public string comarca { get; set; }
19         public bool borrado { get; set; }
20
21         public virtual ICollection<actividad_beneficiario> actividad_beneficiario { get; set; }
22     }
23 }

```

```

01 namespace Entidadees
02 {
03     using System;
04     using System.Collections.Generic;
05
06     public partial class cargos
07     {
08         public cargos()
09         {
10             this.personal_actividad = new HashSet<personal_actividad>();
11         }
12
13         public int id_cargo { get; set; }
14         public string nombre { get; set; }
15         public string descripcion { get; set; }
16         public bool activo { get; set; }
17         public bool borrado { get; set; }
18
19         public virtual ICollection<personal_actividad> personal_actividad { get; set; }
20     }
21 }

```

```

01 namespace Entidadees
02 {
03     using System;
04     using System.Collections.Generic;
05
06     public partial class categoria_gastos
07     {
08         public categoria_gastos()
09         {
10             this.gastos = new HashSet<gastos>();
11         }
12
13         public int id_categoria { get; set; }
14         public string nombre { get; set; }
15         public bool borrado { get; set; }
16
17         public virtual ICollection<gastos> gastos { get; set; }
18     }
19 }

```

```

01 namespace Entidadees

```

```

02 {
03     using System;
04     using System.Collections.Generic;
05
06     public partial class configuracion_cuenta
07     {
08         public int id_cuenta { get; set; }
09         public Nullable<bool> pasivo { get; set; }
10         public Nullable<bool> pasivo_no_corriente { get; set; }
11         public Nullable<bool> activo { get; set; }
12         public Nullable<bool> activo_no_corriente { get; set; }
13
14         public virtual cuenta cuenta { get; set; }
15     }
16 }

01 namespace Entidadees
02 {
03     using System;
04     using System.Collections.Generic;
05
06     public partial class cuenta
07     {
08         public cuenta()
09         {
10             this.diario = new HashSet<diario>();
11         }
12
13         public int id { get; set; }
14         public Nullable<int> id_padre { get; set; }
15         public Nullable<int> nivel { get; set; }
16         public string codigo { get; set; }
17         public string nombre { get; set; }
18         public Nullable<decimal> debito { get; set; }
19         public Nullable<decimal> credito { get; set; }
20         public bool borrado { get; set; }
21
22         public virtual configuracion_cuenta configuracion_cuenta { get; set; }
23         public virtual ICollection<diario> diario { get; set; }
24     }
25 }

```

```

01 namespace Entidadees
02 {
03     using System;
04     using System.Collections.Generic;
05
06     public partial class detalle_gastos
07     {
08         public detalle_gastos()
09         {
10             this.diario = new HashSet<diario>();
11         }
12
13         public int id_detalle_gasto { get; set; }
14         public int id_gastos { get; set; }
15         public int id_entidad_x_actividad { get; set; }
16         public int id_proyecto { get; set; }
17         public string elemento { get; set; }
18         public Nullable<decimal> cantidad { get; set; }
19         public Nullable<decimal> costo_unitario { get; set; }

```

```

20     public Nullable<decimal> iva { get; set; }
21     public Nullable<decimal> descuento { get; set; }
22     public bool borrado { get; set; }
23
24     public virtual gastos gastos { get; set; }
25     public virtual entidad_x_actividad entidad_x_actividad { get; set; }
26     public virtual ICollection<diario> diario { get; set; }
27 }
28 }

01 namespace Entidadees
02 {
03     using System;
04     using System.Collections.Generic;
05
06     public partial class diario
07     {
08         public int id_diario { get; set; }
09         public Nullable<int> id_detalle_gasto { get; set; }
10         public int id_cuenta { get; set; }
11         public decimal debe { get; set; }
12         public decimal haber { get; set; }
13         public bool es_activo { get; set; }
14         public bool borrado { get; set; }
15         public int id_gasto { get; set; }
16         public int id_diario_encabezado { get; set; }
17
18         public virtual cuenta cuenta { get; set; }
19         public virtual detalle_gastos detalle_gastos { get; set; }
20         public virtual diario_encabezado diario_encabezado { get; set; }
21     }
22 }

01 namespace Entidadees
02 {
03     using System;
04     using System.Collections.Generic;
05
06     public partial class donacion_beneficiario
07     {
08         public int id { get; set; }
09         public int id_actividad_beneficiario { get; set; }
10         public decimal cantidad { get; set; }
11         public string observacion { get; set; }
12         public bool borrado { get; set; }
13         public Nullable<int> id_entidad { get; set; }
14
15         public virtual actividad_beneficiario actividad_beneficiario { get; set; }
16     }
17 }

01 namespace Entidadees
02 {
03     using System;
04     using System.Collections.Generic;
05
06     public partial class entidad
07     {
08         public entidad()
09         {
10             this.entidad_tipo_entidad = new HashSet<entidad_tipo_entidad>();
11             this.entidad_x_proyectos = new HashSet<entidad_x_proyectos>();
12         }
13
14         public int id_entidad { get; set; }
15         public string nombre { get; set; }
16         public string siglas { get; set; }
17         public string telefono { get; set; }
18         public string direccion { get; set; }
19         public string correo { get; set; }
20         public bool activo { get; set; }
21         public bool es_empresa { get; set; }
22         public bool borrado { get; set; }

```

```

23         public int id_nacionalidad { get; set; }
24
25         public virtual ICollection<entidad_tipo_entidad> entidad_tipo_entidad { get; set; }
26         public virtual ICollection<entidad_x_proyectos> entidad_x_proyectos { get; set; }
27         public virtual nacionalidades nacionalidades { get; set; }
28     }
29 }

```

```

01 namespace Entidadees
02 {
03     using System;
04     using System.Collections.Generic;
05
06     public partial class entidad_x_actividad
07     {
08         public entidad_x_actividad()
09         {
10             this.personal_actividad = new HashSet<personal_actividad>();
11             this.detalle_gastos = new HashSet<detalle_gastos>();
12         }
13
14         public int id { get; set; }
15         public int id_entidad_x_proyectos { get; set; }
16         public int id_actividad { get; set; }
17         public Nullable<decimal> subcosto { get; set; }
18         public string observaciones { get; set; }
19         public bool es_entrada { get; set; }
20         public bool borrado { get; set; }
21
22         public virtual ICollection<personal_actividad> personal_actividad { get; set; }
23         public virtual entidad_x_proyectos entidad_x_proyectos { get; set; }
24         public virtual actividad actividad { get; set; }
25         public virtual ICollection<detalle_gastos> detalle_gastos { get; set; }
26     }
27 }

```

```

01 namespace Entidadees
02 {
03     using System;
04     using System.Collections.Generic;
05
06     public partial class entidad_x_proyectos
07     {
08         public entidad_x_proyectos()
09         {
10             this.entidad_x_actividad = new HashSet<entidad_x_actividad>();
11             this.entidad_donaciones = new HashSet<entidad_donaciones>();
12         }
13
14         public int id { get; set; }
15         public int id_entidad { get; set; }
16         public int id_proyecto { get; set; }
17         public bool borrado { get; set; }
18
19         public virtual ICollection<entidad_x_actividad> entidad_x_actividad { get; set; }
20         public virtual ICollection<entidad_donaciones> entidad_donaciones { get; set; }
21         public virtual entidad entidad { get; set; }
22         public virtual proyecto proyecto { get; set; }
23     }
24 }

```

```

01 namespace Entidadees
02 {
03     using System;
04     using System.Collections.Generic;
05
06     public partial class proyecto
07     {
08         public proyecto()
09         {
10             this.entidad_x_proyectos = new HashSet<entidad_x_proyectos>();
11         }
12
13         public int id_proyecto { get; set; }

```

```
14     public string nombre { get; set; }
15     public string Objetivo { get; set; }
16     public System.DateTime fecha_inicio { get; set; }
17     public System.DateTime fecha_fin { get; set; }
18     public string codigo { get; set; }
19     public int id_tipo_proyecto { get; set; }
20     public bool finalizado { get; set; }
21     public bool borrado { get; set; }
22     public decimal monto { get; set; }
23
24     public virtual ICollection<Entidad_x_proyectos> entidad_x_proyectos { get; set; }
25     public virtual tipo_proyecto tipo_proyecto { get; set; }
26 }
27 }
```

```

001 using System;
002 using System.Collections.Generic;
003 using System.Linq;
004 using System.Text;
005 using System.Threading.Tasks;
006 using Entidadees;
007 using System.Data.Entity;
008 using System.Web.UI.WebControls;
009
010 namespace GestionProyectos {
011     public class BIActividades {
012
013         public enum TiposActividades {
014             Padres_e_Hijos = 0,
015             Padres = 1,
016             Hijos = 2,
017             Hijos_de_Padre = 3
018         }
019
020         public enum TiposEntidadesActividadMontos {
021             EntradasSalidas = 0,
022             Entradas = 1,
023             Salidas = 2
024         }
025
026 #region CRUD
027
028     public actividad add(actividad toobj)
029     {
030         using (var dbo = new ONGEntities()) {
031             dbo.actividad.Add(toobj);
032             dbo.SaveChanges();
033             return toobj;
034         }
035     }
036
037     public actividad update(actividad toobj)
038     {
039         using (var dbo = new ONGEntities()) {
040             dbo.Entry(toobj).State = EntityState.Modified;
041             dbo.SaveChanges();
042             return toobj;
043         }
044     }
045
046     public bool delete(int id)
047     {
048         using (var dbo = new ONGEntities()) {

```

```

049         var toobj = dbo.actividad.Where(x => x.id_actividad == id).SingleOrDefault();
050         if (toobj != null) {
051             dbo.actividad.Remove(toobj);
052             dbo.SaveChanges();
053             return true;
054         }
055         return false;
056     }
057 }
058
059 public actividad GetById(int id)
060 {
061     using (var dbo = new ONGEntities()) {
062         return dbo.actividad.Where(x => x.id_actividad == id).SingleOrDefault();
063     }
064 }
065
066 public List<actividad> getList()
067 {
068     using (var dbo = new ONGEntities()) {
069         return dbo.actividad.ToList();
070     }
071 }
072 }
073
074 #endregion
075
076 #region LLenado de Objetos por Referencia
077
078     /// <summary>
079     /// Lista las actividades omitiendo el id de la Activida pasado,si es cero se muestran todas,caso para id Activida Padre
080     /// </summary>
081     /// <param name="combox"></param>
082     /// <param name="idActividaOmitir"></param>
083     /// <param name="idProyecto"></param>
084     /// <param name="proyectoEst">Obtener las actividades de acuerdo a un estado de proyecto</param>
085     /// <returns></returns>
086     public static List<Object> loadListaActividades(ref DropDownList combox, int idActividaOmitir = 0, int idProyecto = 0,
087     EstadosGestion.ActividadesProyectoEstado proyectoEst = EstadosGestion.ActividadesProyectoEstado.ambos)
088     {
089         using (var dbo = new ONGEntities()) {
090             var lista = (from ea in dbo.entidad_x_actividad
091                         where ea.entidad_x_proyectos.proyecto.borrado == false && ea.entidad_x_proyectos.borrado == false &&
092                         ea.borrado == false && ea.actividad.borrado == false
093                         && ea.id_actividad != idActividaOmitir
094                         && ea.entidad_x_proyectos.id_proyecto == (idProyecto == 0 ? ea.entidad_x_proyectos.id_proyecto
095                         : idProyecto )

```



```

093         && ea.entidad_x_proyectos.proyecto.finalizado == (proyectoEst ==
EstadosGestion.ActividadesProyectoEstado.finalizados ? true : (proyectoEst == EstadosGestion.ActividadesProyectoEstado.noFinalizados ?
false : ea.entidad_x_proyectos.proyecto.finalizado))
094         group ea.actividad by new { id_actividad = (int?) ea.actividad.id_actividad, ea.actividad.descripcion
} into g
095         select new { id_actividad = (int?) g.Key.id_actividad, g.Key.descripcion }).ToList();
096         combox.DataSource = lista.ToList();
097         combox.DataValueField = "id_actividad";
098         combox.DataTextField = "descripcion";
099         combox.DataBind();
100         combox.Items.Insert(0, new ListItem("*** Seleccione ***", "0"));
101         return lista.ToList<Object>();
102     }
103 }
104
105 /// <summary>
106 /// Lista las actividades omitiendo el id de la Activida pasado, si es cero se muestran todas, caso para id Activida Padre
107 /// </summary>
108 /// <param name="combox"></param>
109 /// <param name="idActividad"></param>
110 /// <param name="idProyecto"></param>
111 /// <returns></returns>
112 public static List<Object> loadListaActividadesPorProyecto(ref DropDownList combox, int idActividad = 0, int idProyecto =
0)
113 {
114     using (var dbo = new ONGEntities()) {
115         var lista = (from ea in dbo.entidad_x_actividad
116                     where ea.entidad_x_proyectos.proyecto.borrado == false && ea.entidad_x_proyectos.borrado == false &&
ea.borrado == false && ea.actividad.borrado == false
117                     && ea.id_actividad == (idActividad == 0 ? ea.id_actividad : idActividad)
118                     && ea.entidad_x_proyectos.id_proyecto == (idProyecto == 0 ? ea.entidad_x_proyectos.id_proyecto :
idProyecto)
119                     group ea.actividad by new { id_actividad = (int?) ea.actividad.id_actividad, ea.actividad.descripcion }
into g
120                     select new { id_actividad = (int?) g.Key.id_actividad, g.Key.descripcion }).ToList();
121         combox.DataSource = lista;
122         combox.DataValueField = "id_actividad";
123         combox.DataTextField = "descripcion";
124         combox.DataBind();
125         combox.Items.Insert(0, new ListItem("*** Seleccione ***", "0"));
126         return lista.ToList<Object>();
127     }
128 }
129
130 /// <summary>
131 /// Obtener las entidades por actividad , Definir el tipo que se requiere a datos
132 /// </summary>
133 /// <param name="idActividad"></param>

```

```

134      /// <param name="tipoEntidadActividad">Si es 0 >>Devuelve todas. 1>> Devuelve las Entradas, 2>> Devuelve las
Salidas</param>
135      /// <returns></returns>
136      public static List<Object> loadListaEntidadesPorActividad(ref DropDownList combobox,int idActividad = 0,
TiposEntidadesActividadMontos tipoEntidadActividad = TiposEntidadesActividadMontos.EntradasSalidas)
137      {
138          using (var dbo = new ONGEntities()) {
139              var lista = (from ea in dbo.entidad_x_actividad
140                          where ea.borrado == false && ea.entidad_x_proyectos.borrado == false &&
141                                ea.entidad_x_proyectos.entidad.borrado == false && ea.es_entrada == false &&
142                                ea.id_actividad == (idActividad == 0 ? ea.id_actividad : idActividad) &&
143                                ea.es_entrada == ((tipoEntidadActividad == TiposEntidadesActividadMontos.EntradasSalidas) ?
ea.es_entrada
144
145                                : (tipoEntidadActividad == TiposEntidadesActividadMontos.Entradas ? true : false))
146                                select new {
147                                    id_entidad_actividad = ea.id,
148                                    entidad = ea.entidad_x_proyectos.entidad.nombre,
149                                    montoCosto = ea.subcosto,
150                                    esEntrada = (ea.es_entrada ? "SI" : "NO"),
151                                }).ToList<Object>();
152
153              combobox.DataSource = lista;
154              combobox.DataValueField = "id_entidad_actividad";
155              combobox.DataTextField = "entidad";
156              combobox.DataBind();
157              combobox.Items.Insert(0, new ListItem("*** Seleccione ***", "0"));
158              return lista;
159          }
160
161      /// <summary>
162      /// Obtener las entidades por actividad , Definir el tipo que se requiere a datos
163      /// </summary>
164      /// <param name="idActividad"></param>
165      /// <param name="tipoEntidadActividad">Si es 0 >>Devuelve todas. 1>> Devuelve las Entradas, 2>> Devuelve las
Salidas</param>
166      /// <returns></returns>
167      public static List<Object> loadListaEntidadesPorActividad2(ref DropDownList combobox, int idActividad = 0,
TiposEntidadesActividadMontos tipoEntidadActividad = TiposEntidadesActividadMontos.EntradasSalidas)
168      {
169          using (var dbo = new ONGEntities()) {
170              var lista = (from ea in dbo.entidad_x_actividad
171                          where ea.borrado == false && ea.entidad_x_proyectos.borrado == false &&
172                                ea.entidad_x_proyectos.entidad.borrado == false && ea.es_entrada == false &&
173                                ea.id_actividad == (idActividad == 0 ? ea.id_actividad : idActividad) &&
174                                ea.es_entrada == ((tipoEntidadActividad == TiposEntidadesActividadMontos.EntradasSalidas) ?
ea.es_entrada
175
176                                : (tipoEntidadActividad == TiposEntidadesActividadMontos.Entradas ? true : false))

```

```

176         select new {
177             id_entidad_actividad = ea.id,
178             entidad = ea.entidad_x_proyectos.entidad.nombre + " - Es Entrada: " + (ea.es_entrada ? "SI" :
"NO"),
179             montoCosto = ea.subcosto,
180             esEntrada = (ea.es_entrada ? "SI" : "NO"),
181         }).ToList<Object>();
182
183     combo.DataSource = lista;
184     combo.DataValueField = "id_entidad_actividad";
185     combo.DataTextField = "entidad";
186     combo.DataBind();
187     combo.Items.Insert(0, new ListItem("*** Seleccione ***", "0"));
188     return lista;
189 }
190 }
191
192 #endregion
193
194 #region Logica
195
196     /// <summary>
197     /// Obtener el proyecto de la actividad
198     /// </summary>
199     /// <param name="idActividad"></param>
200     /// <returns></returns>
201     public static proyecto getProyectoDeActividad(int idActividad){
202         using (var dbo = new ONGEntities()) {
203             var dato = (from ea in dbo.entidad_x_actividad
204                 join ep in dbo.entidad_x_proyectos on ea.id_entidad_x_proyectos equals ep.id
205                 join p in dbo.proyecto on ep.id_proyecto equals p.id_proyecto
206                 where ea.borrado == false && ep.borrado == false && ea.id_actividad == idActividad
207                 select p).ToList();
208             return (dato.Count > 0 ? dato[0] : null);
209         }
210     }
211
212     /// <summary>
213     /// Obtener la lista los tipos de actividades, actividades hijos o padres, tambien parametrizado por un id
214     /// </summary>
215     /// <param name="idActividad">Id de la actividad, si es cero devuelve todos</param>
216     /// <param name="idProyecto">Filtrar por id del proyecto , si se requiere</param>
217     /// <param name="proyectoEst">Obtener las actividades de acuerdo al estado del proyecto</param>
218     /// <returns></returns>
219     public List<Object> getListaActividades(int idActividad = 0, int idProyecto = 0, EstadosGestion.ActividadesProyectoEstado
proyectoEst = EstadosGestion.ActividadesProyectoEstado.ambos)
220     {
221         using (var dbo = new ONGEntities()) {

```

```

222         var lista = (from a in dbo.actividad join po in
223             ( (from ea in dbo.entidad_x_actividad
224                 where ea.entidad_x_proyectos.proyecto.borrado == false &&
ea.entidad_x_proyectos.borrado == false && ea.borrado == false
225                 select new {
226                     id_proyecto = (int?)ea.entidad_x_proyectos.proyecto.id_proyecto,
227                     proyecto = ea.entidad_x_proyectos.proyecto.nombre,
228                     ea.id_actividad,
229                     proyectoFinalizado = ea.entidad_x_proyectos.proyecto.finalizado
230                 }) on new { id_actividad = a.id_actividad } equals new { id_actividad =
(int)po.id_actividad }
231             where a.borrado == false && a.id_actividad == (idActividad == 0 ? a.id_actividad :
idActividad)
232             && po.id_proyecto == (idProyecto == 0 ? po.id_proyecto : idProyecto)
233             && po.proyectoFinalizado == (proyectoEst ==
EstadosGestion.ActividadesProyectoEstado.finalizados ? true : (proyectoEst == EstadosGestion.ActividadesProyectoEstado.noFinalizados ?
false : po.proyectoFinalizado))
234             orderby a.descripcion
235             select new {
236                 id_actividad = a.id_actividad,
237                 fecha = a.fecha,
238                 fecha_fin = a.fecha_fin,
239                 actividad = a.descripcion,
240                 codigoActividad = a.codigo,
241                 observacion = a.observacion,
242                 idActividaPadre = a.idActividaPadre,
243                 borrado = a.borrado,
244                 id_proyecto = po.id_proyecto,
245                 proyecto = po.proyecto,
246                 finalizado = a.finalizado,
247                 padreDescripcion = ((from aa in dbo.actividad where aa.id_actividad == a.idActividaPadre
select new { aa.descripcion }).FirstOrDefault().descripcion)
248                 }).Distinct().ToList<Object>();
249
250         return lista;
251     }
252 }
253
254 /// <summary>
255 /// Obtener el Proyecto y Actividad relacionad al Gasto
256 /// </summary>
257 /// <param name="idGasto"></param>
258 /// <param name="idProyecto"></param>
259 /// <param name="idActividad"></param>
260 /// <returns></returns>
261 public static List<Object> getProyectoAndActividadGasto(int idGasto, out int idProyecto, out int idActividad) {
262     using (var dbo = new ONGEntities()) {
263         var datos = (from dg in dbo.detalle_gastos

```

```

264         where
265             dg.entidad_x_actividad.borrado == false &&
266             dg.entidad_x_actividad.entidad_x_proyectos.borrado == false &&
267             dg.entidad_x_actividad.entidad_x_proyectos.entidad.borrado == false &&
268             dg.entidad_x_actividad.entidad_x_proyectos.proyecto.borrado == false &&
269             dg.id_gastos == idGasto
270         select new {
271             id_proyecto = (int?)dg.entidad_x_actividad.entidad_x_proyectos.proyecto.id_proyecto,
272             proyecto = dg.entidad_x_actividad.entidad_x_proyectos.proyecto.nombre,
273             id_actividad = (int?)dg.entidad_x_actividad.actividad.id_actividad,
274             actividad = dg.entidad_x_actividad.actividad.descripcion
275         }).ToList();
276
277         if (datos.Count() > 0) {
278             idProyecto = Convert.ToInt32(datos[0].id_proyecto);
279             idActividad = Convert.ToInt32(datos[0].id_actividad);
280         }
281         else {
282             idProyecto = 0; idActividad = 0;
283         }
284         return datos.ToList<Object>();
285     }
286 }
287
288
289 /// <summary>
290 /// Guardar la actividad y sus correspondientes entidades
291 /// </summary>
292 /// <param name="tooActividad"></param>
293 /// <param name="listaEntidadesNuevas"></param>
294 /// <param name="idONG"></param>
295 /// <param name="montoONG"></param>
296 /// <returns></returns>
297 public actividad guardarActividadAndEntidades(actividad tooActividad, List<entidad_x_actividad> listaEntidadesNuevas)
298 {
299     using (var dbo = new ONGEntities())
300     {
301         if (tooActividad.idActividaPadre == 0) { tooActividad.idActividaPadre = null; }
302
303         if (tooActividad.id_actividad == 0) {
304             this.add(tooActividad);
305
306             /** Generar el codigo de la actividad con el id */
307             string codigoActividad = "a-" + tooActividad.id_actividad.ToString("00#");
308             dbo.actividad.Where(x => x.id_actividad == tooActividad.id_actividad).ToList().ForEach(x => x.codigo =
codigoActividad);
309         }
310         else {

```

```

311         this.update(tooActividad);
312     }
313
314     foreach (var item in listaEntidadesNuevas) {
315         item.borrado = false;
316         item.id_actividad = tooActividad.id_actividad;
317         dbo.entidad_x_actividad.Add(item);
318     }
319
320     dbo.SaveChanges();
321     return tooActividad;
322 }
323
324
325 /// <summary>
326 /// Cambiar de estado una entidad en la actividad
327 /// </summary>
328 /// <param name="idEntidadActividad"></param>
329 /// <returns></returns>
330 public bool deleteEntidadDeActividad(int idEntidadActividad)
331 {
332     using (var dbo = new ONGEntities()) {
333         var too = dbo.entidad_x_actividad.Where(x => x.id == idEntidadActividad).SingleOrDefault();
334         too.borrado = true;
335         dbo.Entry(too).State = EntityState.Modified;
336         dbo.SaveChanges();
337         return true;
338     }
339 }
340
341 /// <summary>
342 /// Obtener las entidades por actividad , Definir el tipo que se requiere a datos
343 /// </summary>
344 /// <param name="idActividad"></param>
345 /// <param name="tipoEntidadActividad">Si es 0 >>Devuelve todas. 1>> Devuelve las Entradas, 2>> Devuelve las
346 Salidas</param>
347 /// <returns></returns>
348 public List<Object> getListEntidadesPorActividad(int idActividad, TiposEntidadesActividadMontos tipoEntidadActividad)
349 {
350     using (var dbo = new ONGEntities())
351     {
352         var lista = (from ea in dbo.entidad_x_actividad
353                     where ea.borrado == false && ea.entidad_x_proyectos.borrado == false &&
354                           ea.entidad_x_proyectos.entidad.borrado == false &&
355                           ea.id_actividad == (idActividad == 0 ? ea.id_actividad : idActividad) &&
356                           ea.es_entrada == ((tipoEntidadActividad == TiposEntidadesActividadMontos.EntradasSalidas) ?
357                                           : (tipoEntidadActividad == TiposEntidadesActividadMontos.Entradas ? true : false))

```

```

357         select new {
358             id_entidad_actividad = ea.id,
359             id_entidad_proyecto = ea.id_entidad_x_proyectos,
360             entidad = ea.entidad_x_proyectos.entidad.nombre,
361             montoCosto = ea.subcosto,
362             esEntrada = (ea.es_entrada ? "SI" : "NO"),
363             observaciones = ea.observaciones
364         }).ToList<Object>();
365     return lista;
366 }
367 }
368
369 /// <summary>
370 /// Obtener las entidades por actividad , Definir el tipo que se requiere a datos
371 /// </summary>
372 /// <param name="idActividad"></param>
373 /// <param name="tipoEntidadActividad">Si es 0 >>Devuelve todas. 1>> Devuelve las Entradas, 2>> Devuelve las
Salidas</param>
374 /// <returns></returns>
375 public List<Object> getListaEntidadesPorActividad2(int idActividad, TiposEntidadesActividadMontos tipoEntidadActividad)
376 {
377     using (var dbo = new ONGEntities()) {
378         var lista = (from ea in dbo.entidad_x_actividad
379             where ea.borrado == false && ea.entidad_x_proyectos.borrado == false &&
380                 ea.entidad_x_proyectos.entidad.borrado == false &&
381                 ea.id_actividad == (idActividad == 0 ? ea.id_actividad : idActividad) &&
382                 ea.es_entrada == ((tipoEntidadActividad == TiposEntidadesActividadMontos.EntradasSalidas) ?
383                     : (tipoEntidadActividad == TiposEntidadesActividadMontos.Entradas ? true : false))
384             select new {
385                 id_entidad_actividad = ea.id,
386                 id_entidad_proyecto = ea.id_entidad_x_proyectos,
387                 entidad = ea.entidad_x_proyectos.entidad.nombre + " - Es Entrada: " + (ea.es_entrada ? "SI" :
"NO"),
388                 montoCosto = ea.subcosto,
389                 esEntrada = (ea.es_entrada ? "SI" : "NO"),
390                 observaciones = ea.observaciones
391             }).ToList<Object>();
392     return lista;
393 }
394 }
395
396 /// <summary>
397 /// Agregar o Modificar un Personal Actividad a la Actividad
398 /// </summary>
399 /// <param name="toob"></param>
400 /// <returns></returns>
401 public personal_actividad guardarPesonalActividad(personal_actividad toob) {

```

```

402         using(var dbo = new ONGEntities()) {
403             if(toob.id == 0) {
404                 toob.borrado = false;
405                 dbo.personal_actividad.Add(toob);
406             }
407             else {
408                 dbo.Entry(toob).State = EntityState.Modified;
409             }
410             dbo.SaveChanges();
411             return toob;
412         }
413     }
414
415     /// <summary>
416     /// Pasar a borrado la actividad y el personal actividad
417     /// </summary>
418     /// <param name="idActividad"></param>
419     /// <returns></returns>
420     public bool borrarActividadAndPersonalActividad(int idActividad) {
421         using (var dbo = new ONGEntities()) {
422             var tooActi = dbo.actividad.Where(z => z.id_actividad == idActividad).SingleOrDefault();
423             if (tooActi != null) {
424                 var listEntidadActividad = dbo.entidad_x_actividad.Where(x => x.id_actividad == idActividad).ToList();
425
426                 /*** Verificar si tiene personal entidad la lista de valores de la entidad por actividad ***/
427                 foreach (var item in listEntidadActividad) {
428                     var listaPesonaleA = dbo.personal_actividad.Where(x => x.id_entidad_x_actividad == item.id).ToList();
429                     if (listEntidadActividad.Count() > 0){
430                         listaPesonaleA.ForEach(x => x.borrado = true);
431                     }
432                     item.borrado = true;
433                     dbo.Entry(item).State = EntityState.Modified;
434                 }
435
436                 tooActi.borrado = true;
437                 dbo.Entry(tooActi).State = EntityState.Modified;
438                 dbo.SaveChanges();
439                 return true;
440             }
441             return false;
442         }
443     }
444
445 #endregion
446
447 }
448 }

```



```

001 using System;
002 using System.Collections.Generic;
003 using System.Linq;
004 using System.Text;
005 using System.Threading.Tasks;
006 using Entidadees;
007 using System.Data.Entity;
008
009 namespace GestionProyectos {
010     public class BLGastos {
011
012         #region Metodos CRUD
013
014         public gastos add(gastos toobj)
015         {
016             using (var dbo = new ONGEntities()) {
017                 dbo.gastos.Add(toobj);
018                 dbo.SaveChanges();
019                 return toobj;
020             }
021         }
022
023         public gastos update(gastos toobj)
024         {
025             using (var dbo = new ONGEntities()) {
026                 dbo.Entry(toobj).State = EntityState.Modified;
027                 dbo.SaveChanges();
028                 return toobj;
029             }
030         }
031
032         public bool delete(int id)
033         {
034             using (var dbo = new ONGEntities()) {
035                 var toobj = dbo.gastos.Where(x => x.id_gastos == id).SingleOrDefault();
036                 if (toobj != null) {
037                     dbo.gastos.Remove(toobj);
038                     dbo.SaveChanges();
039                     return true;
040                 }
041                 return false;
042             }
043         }
044
045         public gastos getById(int id)
046         {
047             using (var dbo = new ONGEntities()) {

```

```

048         var toobj = dbo.gastos.Where(x => x.id_gastos == id).SingleOrDefault();
049         return toobj;
050     }
051 }
052
053 public List<gastos> getList()
054 {
055     using (var dbo = new ONGEntities()) {
056         return dbo.gastos.ToList();
057     }
058 }
059
060 #endregion
061
062 #region Logica Negocio
063
064     /// <summary>
065     /// Obtener un objeto Gasto (encabezado) por un idActividad
066     /// </summary>
067     /// <param name="idActividad"></param>
068     /// <returns></returns>
069     public gastos getGastoPorActividad(int idActividad)
070     {
071         using (var dbo = new ONGEntities()) {
072             var tolist = (from dg in dbo.detalle_gastos
073                 join g in dbo.gastos on dg.id_gastos equals g.id_gastos
074                 //where dg.id_actividad == idActividad
075                 select new gastos {
076                     id_gastos = g.id_gastos,
077                     id_categoria = g.id_categoria,
078                     factura = g.factura,
079                     recibo = g.recibo,
080                     fecha = g.fecha,
081                     //proveedor = g.proveedor,
082                     IR = g.IR,
083                     tasa_cambio_us = g.tasa_cambio_us,
084                     tasa_cambio_ue = g.tasa_cambio_ue,
085                     iva = g.iva,
086                     descuento = g.descuento,
087                     descripcion = g.descripcion,
088                     subtotal = g.subtotal,
089                     total = g.total
090                 }).ToList();
091             if (tolist.Count > 0) {
092                 var toobj = new gastos();
093                 toobj = tolist.SingleOrDefault();
094                 return toobj;
095             }

```

```

096         return null;
097     }
098 }
099
100     /// <summary>
101     /// Obtener la lista de Cabeceras de Gastos junto con la actividad
102     /// </summary>
103     /// <param name="idActividad">Si es cero, devuelve todo, si no el valor que se establecec</param>
104     /// <param name="idGasto">Si es cero, devuelve todo, si no el valor que se establecec</param>
105     /// <param name="proyectoEst">Obtener los gastos de acuerdo a los proyectos con estados parametrizados</param>
106     /// <returns></returns>
107     public List<Object> getListaGastosActividad(int idActividad = 0, int idGasto = 0, EstadosGestion.GastosProyectoEstado
proyectoEst = EstadosGestion.GastosProyectoEstado.ambos)
108     {
109         using (var dbo = new ONGEntities()) {
110
111             var lista = (from dg in dbo.detalle_gastos
112                 where
113                     dg.entidad_x_actividad.entidad_x_proyectos.proyecto.borrado == false &&
114                     dg.entidad_x_actividad.entidad_x_proyectos.borrado == false &&
115                     dg.entidad_x_actividad.borrado == false &&
116                     dg.borrado == false &&
117                     dg.gastos.borrado == false &&
118                     dg.gastos.proveedores.borrado == false &&
119                     dg.gastos.id_gastos == (idGasto == 0 ? dg.gastos.id_gastos : idGasto) &&
120                     dg.entidad_x_actividad.actividad.id_actividad == (idActividad == 0 ?
dg.entidad_x_actividad.actividad.id_actividad : idActividad)
121                 group new { dg.entidad_x_actividad.entidad_x_proyectos.proyecto, dg.entidad_x_actividad.actividad,
dg.gastos, dg.gastos.proveedores } by new {
122                     dg.entidad_x_actividad.entidad_x_proyectos.proyecto.nombre,
123                     proyectoFinalizado = dg.entidad_x_actividad.entidad_x_proyectos.proyecto.finalizado,
124                     dg.entidad_x_actividad.actividad.descripcion,
125                     actividadFinalizada = dg.entidad_x_actividad.actividad.finalizado,
126                     id_gastos = (int?)dg.gastos.id_gastos,
127                     codigoGasto = dg.gastos.codigo,
128                     dg.gastos.factura,
129                     dg.gastos.recibo,
130                     fecha = (DateTime?)dg.gastos.fecha,
131                     dg.gastos.proveedores.proveedor,
132                     subtotal = (decimal?)dg.gastos.subtotal,
133                     iva = (decimal?)dg.gastos.iva,
134                     descuento = (decimal?)dg.gastos.descuento,
135                     total = (decimal?)dg.gastos.total,
136                     tasa_cambio_us = (decimal?)dg.gastos.tasa_cambio_us,
137                     tasa_cambio_ue = (decimal?)dg.gastos.tasa_cambio_ue,
138                     IR = (decimal?)dg.gastos.IR ,
139                     es_vinculado = dg.gastos.es_vinculado
140                 } into g

```

```

141         select new {
142             proyecto = g.Key.nombre,
143             g.Key.proyectoFinalizado,
144             actividad = g.Key.descripcion,
145             g.Key.codigoGasto,
146             g.Key.actividadFinalizada,
147             id_gastos = (int?)g.Key.id_gastos,
148             g.Key.factura,
149             g.Key.recibo,
150             fecha = (DateTime?)g.Key.fecha,
151             g.Key.proveedor,
152             subtotal = (decimal?)g.Key.subtotal,
153             iva = (decimal?)g.Key.iva,
154             descuento = (decimal?)g.Key.descuento,
155             total = (decimal?)g.Key.total,
156             tasa_cambio_us = (decimal?)g.Key.tasa_cambio_us,
157             tasa_cambio_ue = (decimal?)g.Key.tasa_cambio_ue,
158             IR = (decimal?)g.Key.IR,
159             es_vinculado = g.Key.es_vinculado
160         }).Where(y => y.proyectoFinalizado == (proyectoEst ==
EstadosGestion.GastosProyectoEstado.finalizados ? true : (proyectoEst == EstadosGestion.GastosProyectoEstado.noFinalizados ? false :
y.proyectoFinalizado))).ToList<Object>();
161         return lista;
162     }
163 }
164
165 /// <summary>
166 /// Obtener un objeto gasto por idActividad
167 /// </summary>
168 /// <param name="idActividad"></param>
169 /// <returns></returns>
170 public gastos getGastoByIdActividad(int idActividad)
171 {
172     using (var dbo = new ONGEntities()) {
173         var toobj = dbo.gastos.Where(x => x.id_actividad == idActividad).ToList();
174         if (toobj.Count > 0)
175             return toobj[0];
176         return null;
177     }
178 }
179
180 /// <summary>
181 /// Guardar un gatos con su actividad, si ya existe gasto se actualiza los detalles
182 /// </summary>
183 /// <param name="idProyecto"></param>
184 /// <param name="idActividad"></param>
185 /// <param name="idGasto"></param>
186 /// <param name="idCategoriaGasto"></param>

```

```

187      /// <param name="proveedor"></param>
188      /// <param name="factura"></param>
189      /// <param name="recibo"></param>
190      /// <param name="fecha"></param>
191      /// <param name="tcUS"></param>
192      /// <param name="tcUE"></param>
193      /// <param name="ir"></param>
194      /// <param name="item"></param>
195      /// <param name="cantidad"></param>
196      /// <param name="costoUnitario"></param>
197      /// <param name="idCuentaContable"></param>
198      /// <param name="ivaCabecera"></param>
199      /// <param name="ivaDetalle"></param>
200      /// <param name="descuentoCabecera"></param>
201      /// <param name="descuentoDetalle"></param>
202      /// <param name="subtotalCabecera"></param>
203      /// <param name="totalCabecera"></param>
204      /// <param name="descripcion"></param>
205      /// <returns></returns>
206      public detalle_gastos guardarGastodeActividad(int idProyecto, int idActividad, int idGasto, int idCategoriaGasto,
string proveedor, string factura, string recibo, DateTime fecha, decimal tcUS, decimal tcUE, decimal ir,
207      string item, decimal cantidad, decimal costoUnitario, int idCuentaContable, decimal
ivaCabecera, decimal ivaDetalle, decimal descuentoCabecera, decimal descuentoDetalle, decimal subtotalCabecera, decimal totalCabecera,
208      string descripcion)
209      {
210          using (var dbo = new ONGEntities()) {
211              var toGasto = new gastos() {
212                  id_gastos = idGasto,
213                  id_categoria = idCategoriaGasto,
214                  //proveedor = proveedor.Trim(),
215                  factura = (factura == null ? "" : factura),
216                  recibo = (recibo == null ? "" : recibo),
217                  fecha = fecha,
218                  tasa_cambio_us = tcUS,
219                  tasa_cambio_ue = tcUE,
220                  IR = ir,
221                  descripcion = descripcion,
222                  id_actividad = idActividad
223              };
224
225              if (Convert.ToInt32(idGasto) == 0) {
226                  toGasto.subtotal = Convert.ToDecimal(Math.Round(Convert.ToDouble(subtotalCabecera), 2));
227                  toGasto.iva = Convert.ToDecimal(Math.Round(Convert.ToDouble(ivaCabecera), 2));
228                  toGasto.descuento = Convert.ToDecimal(Math.Round(Convert.ToDouble(descuentoCabecera), 2));
229                  toGasto.total = Convert.ToDecimal(Math.Round(Convert.ToDouble(totalCabecera), 2));
230                  dbo.gastos.Add(toGasto);
231              }
232              else if (Convert.ToInt32(idGasto) > 0) {

```

```

233         toGasto = dbo.gastos.Where(x => x.id_gastos == idGasto).SingleOrDefault();
234         toGasto.subtotal = Convert.ToDecimal(Math.Round(Convert.ToDouble(toGasto.subtotal + subtotalCabecera),
235 2)));
236         toGasto.iva = Convert.ToDecimal(Math.Round(Convert.ToDouble(toGasto.iva + ivaCabecera), 2));
237         toGasto.descuento = Convert.ToDecimal(Math.Round(Convert.ToDouble(toGasto.descuento + descuentoCabecera),
238 2)));
239         toGasto.total = Convert.ToDecimal(Math.Round(Convert.ToDouble(toGasto.total + totalCabecera), 2));
240         dbo.Entry(toGasto).State = EntityState.Modified;
241     }
242
243     dbo.SaveChanges();
244
245     if (toGasto.id_gastos != 0) {
246         var toDetalleGasto = new detalle_gastos() {
247             id_proyecto = idProyecto,
248             //id actividad = toGasto.id actividad,
249             id_gastos = toGasto.id_gastos,
250             //id_cuenta = idCuentaContable,
251             elemento = item.Trim(),
252             cantidad = Convert.ToDecimal(Math.Round(Convert.ToDouble(cantidad), 2)),
253             costo_unitario = Convert.ToDecimal(Math.Round(Convert.ToDouble(costoUnitario), 2)),
254             iva = Convert.ToDecimal(Math.Round(Convert.ToDouble(ivaDetalle), 2)),
255             descuento = Convert.ToDecimal(Math.Round(Convert.ToDouble(descuentoDetalle), 2))
256         };
257
258         dbo.detalle_gastos.Add(toDetalleGasto);
259         dbo.SaveChanges();
260
261         if (toGasto.id_actividad != idActividad) {
262             //actualizarProyectoActividadEnDetalle(toGasto.id_actividad, toGasto.id_gastos);
263         }
264         return toDetalleGasto;
265     }
266     return null;
267 }
268
269 /// <summary>
270 /// Borrar un detalle gasto , por id del detalle gasto y actualizar la info del gasto de cabecera
271 /// </summary>
272 /// <param name="idDetalleGasto"></param>
273 /// <returns></returns>
274 public detalle_gastos borrarDetalleGastoById(int idDetalleGasto)
275 {
276     using (var dbo = new ONGEntities()) {
277         var toobj = dbo.detalle_gastos.Where(x => x.id_detalle_gasto == idDetalleGasto).SingleOrDefault();
278         if (toobj != null) {

```

```

279         dbo.detalle_gastos.Where(x => x.id_detalle_gasto == idDetalleGasto).ToList().ForEach(x => x.borrado =
true);
280
281         var toobjGasto = dbo.gastos.Where(z => z.id_gastos == toobj.id_gastos).SingleOrDefault();
282         if (toobjGasto != null) {
283             decimal subtotal = Convert.ToDecimal(toobj.cantidad) * Convert.ToDecimal(toobj.costo_unitario);
284             decimal iva = subtotal * Convert.ToDecimal(toobj.iva / 100);
285             decimal descuento = subtotal * Convert.ToDecimal(toobj.descuento / 100);
286             decimal total = (subtotal - descuento) + iva;
287
288             toobjGasto.subtotal = toobjGasto.subtotal - subtotal;
289             toobjGasto.iva = toobjGasto.iva - iva;
290             toobjGasto.descuento = toobjGasto.descuento - descuento;
291             toobjGasto.total = toobjGasto.total - total;
292             dbo.Entry(toobjGasto).State = EntityState.Modified;
293
294         }
295         dbo.SaveChanges();
296         return toobj;
297     }
298 }
299 return null;
300 }
301
302
303 /// <summary>
304 /// Borrar un Gasto y Su Detalle
305 /// </summary>
306 /// <param name="idGasto"></param>
307 /// <returns></returns>
308 public bool borrarGastoAndDetalle(int idGasto) {
309     using(var dbo = new ONGEntities()) {
310         var listDetallesGasto = dbo.detalle_gastos.Where(x => x.id_gastos == idGasto).ToList();
311         var tooGasto = dbo.gastos.Where(x => x.id_gastos == idGasto).SingleOrDefault();
312
313         if (listDetallesGasto.Count() > 0) {
314             listDetallesGasto.ForEach(x => x.borrado = true);
315         }
316
317         if(tooGasto != null) {
318             tooGasto.borrado = true;
319             dbo.Entry(tooGasto).State = EntityState.Modified;
320         }
321         dbo.SaveChanges();
322         return true;
323     }
324 }
325

```

```

326
327     /// <summary>
328     /// Guardar la informacion del excel , procesar el excel, si los datos de la cabecera del gasto son identicos solo se
guardara el detalle
329     /// </summary>
330     /// <param name="idActividad"></param>
331     /// <param name="categoriaGasto"></param>
332     /// <param name="factura"></param>
333     /// <param name="recibo"></param>
334     /// <param name="fecha"></param>
335     /// <param name="tcUS"></param>
336     /// <param name="tcUE"></param>
337     /// <param name="item"></param>
338     /// <param name="cantidad"></param>
339     /// <param name="costoUnitario"></param>
340     /// <param name="cuentaContable"></param>
341     /// <param name="iva"></param>
342     /// <param name="descuento"></param>
343     /// <param name="subtotal"></param>
344     /// <param name="total"></param>
345     /// <param name="descripcion"></param>
346     /// <returns></returns>
347     public bool guardarSubidaGastosExcel(int idActividad, string categoriaGasto, string factura, string recibo, DateTime
fecha, string proveedor, decimal ir, decimal tcUS, decimal tcUE,
348         string item, decimal cantidad, decimal costoUnitario, string cuentaContable, decimal
iva, decimal descuento, decimal subtotal, string descripcion)
349     {
350         using (var dbo = new ONGEntities()) {
351             //Buscando llaves de datos existentes
352             //
353             //var toCategoriaGasto = dbo.categoria_gastos.Where(x => x.nombre.ToUpper().Trim() ==
categoriaGasto.ToUpper().Trim()).SingleOrDefault();
354             //int idCategoriaGasto = toCategoriaGasto.id_categoria;
355             int idCategoriaGasto = Convert.ToInt32(categoriaGasto);
356
357             var toActividad = dbo.actividad.Where(x => x.id_actividad == idActividad).SingleOrDefault();
358             //int idProyecto = toActividad.id_proyecto;
359             int idProyecto = 0;
360
361
362             var toCuentaContable = dbo.cuenta.Where(x => x.codigo.Trim() == cuentaContable.Trim()).SingleOrDefault();
363             int idCuentaContable = toCuentaContable.id;
364
365             //Buscar el gasto si existe , informacion parecida
366             var toListGasto = (from g in dbo.gastos
367                             join dg in dbo.detalle_gastos on g.id_gastos equals dg.id_gastos
368                             where dg.id_proyecto == idProyecto
369                             //&& dg.id_actividad == idActividad

```



```

370         && g.factura == (factura == null ? "" : factura.Trim())
371         && g.recibo == (recibo == null ? "" : recibo.Trim())
372         //&& g.IR == ir && g.proveedor.Equals(proveedor.Trim())
373         && g.fecha == fecha && g.tasa_cambio_us == tcUS && g.tasa_cambio_ue == tcUE
374     select g).ToList();
375
376     var toGasto = new gastos();
377
378     if (toListGasto.Count <= 0) // Es nuevo encabezado de gasto, se crea el objeto y con el subtotal y total en
379     cero
380     {
381         var toObjeto = new gastos() {
382             id_categoria = idCategoriaGasto,
383             factura = (factura == null ? "" : factura.Trim()),
384             recibo = (recibo == null ? "" : recibo.Trim()),
385             fecha = fecha,
386             //proveedor = proveedor.Trim(),
387             IR = ir,
388             tasa_cambio_us = tcUS,
389             tasa_cambio_ue = tcUE,
390             iva = 0,
391             descuento = 0,
392             subtotal = 0,
393             total = 0,
394             descripcion = descripcion.Trim(),
395             id_actividad = idActividad
396         };
397         toGasto = toObjeto;
398         dbo.gastos.Add(toGasto);
399         dbo.SaveChanges();
400     }
401     else
402     {
403         toGasto = toListGasto[0];
404
405         //Insertar el detalle
406         if (toGasto.id_gastos != 0) {
407             var toDetalleGasto = new detalle_gastos() {
408                 id_proyecto = idProyecto,
409                 //id_actividad = idActividad,
410                 id_gastos = toGasto.id_gastos,
411                 //id_cuenta = idCuentaContable,
412                 elemento = item.Trim(),
413                 cantidad = cantidad,
414                 costo_unitario = Convert.ToDecimal(Math.Round(Convert.ToDouble( costoUnitario),2)),
415                 iva = iva,
416                 descuento = descuento
417             };
418             dbo.detalle_gastos.Add(toDetalleGasto);
419         }
420     }

```

```

417
418         **** Quitar los porcentajes ****/
419         descuento = (subtotal * (descuento / 100));
420         iva = (subtotal * (iva / 100));
421         decimal total = (subtotal + iva) - descuento;
422
423         toGasto = dbo.gastos.Where(x => x.id_gastos == toGasto.id_gastos).SingleOrDefault();
424         toGasto.subtotal = Convert.ToDecimal(Math.Round(Convert.ToDouble(toGasto.subtotal + subtotal),2));
425         toGasto.iva = Convert.ToDecimal(Math.Round(Convert.ToDouble(toGasto.iva + iva),2));
426         toGasto.descuento = Convert.ToDecimal(Math.Round(Convert.ToDouble(toGasto.descuento + descuento),2));
427         toGasto.total = Convert.ToDecimal(Math.Round(Convert.ToDouble(toGasto.total + total), 2));
428         dbo.Entry(toGasto).State = EntityState.Modified;
429         dbo.SaveChanges();
430
431         return true;
432     }
433 }
434 return false;
435 }
436
437
438 /// <summary>
439 /// Obtener Lista del Detalle De los Gastos
440 /// </summary>
441 /// <param name="idGasto">Si el valor es cero devuelve todos, caso contrario devuelve de acuerdo al valor</param>
442 /// <returns></returns>
443 public List<Object> getListaDetalleGastosPorGasto(int idGasto = 0)
444 {
445     using (var dbo = new ONGEntities()) {
446         var lista = (from dg in dbo.detalle_gastos join entidades in (
447             (from ep in dbo.entidad_x_proyectos where ep.entidad.borrado == false && ep.borrado == false
448             select new {
449                 id_entidad_proyectos = ep.id, entidad = ep.entidad.nombre
450             }) on new { id_entidad_proyectos = dg.entidad_x_actividad.id_entidad_x_proyectos } equals
451             new { id_entidad_proyectos = (int) entidades.id_entidad_proyectos }
452             where dg.borrado == false && dg.entidad_x_actividad.borrado == false && dg.id_gastos == (idGasto==
453             0 ? dg.id_gastos : idGasto)
454             select new {
455                 dg.id_detalle_gasto,
456                 id_entidad_actividad = dg.id_entidad_x_actividad,
457                 entidades.entidad,
458                 item = dg.elemento,
459                 dg.cantidad,
460                 dg.costo_unitario,
461                 itemSubtotal = (dg.cantidad * dg.costo_unitario),
462                 itemIva = dg.iva,
463                 itemDescuento = dg.descuento,

```

```

462             itemTotal = (decimal?) (((dg.cantidad * dg.costo_unitario) - ((dg.cantidad *
dg.costo_unitario) * (dg.descuento / 100))) + ((dg.cantidad * dg.costo_unitario) * (dg.iva / 100)))
463         })).ToList<Object>());
464         return lista;
465     }
466 }
467
468
469     /// <summary>
470     /// Guardar un objeto gastos junto con los nuevos items a agregar(no actualizar)
471     /// </summary>
472     /// <param name="toobGastos">Objeto Gasto</param>
473     /// <param name="listaDetallesItemsNews">Lista de nuevos Items ingresados</param>
474     /// <returns></returns>
475     public bool guardarGastoAndNewDetalles(gastos toobGastos, List<detalle_gastos> listaDetallesItemsNews) {
476         using(var dbo = new ONGEntities()) {
477
478             if(toobGastos.id_gastos == 0) {
479                 toobGastos.borrado = false;
480                 this.add(toobGastos);
481
482                 /** Generar el codigo del gasto con el id ***/
483                 string codigoGasto = "g-" + toobGastos.id_gastos.ToString("00#");
484                 dbo.gastos.Where(x => x.id_gastos == toobGastos.id_gastos).ToList().ForEach(x => x.codigo = codigoGasto);
485
486             }else {
487                 this.update(toobGastos);
488             }
489
490             foreach(var item in listaDetallesItemsNews) {
491                 item.borrado = false;
492                 item.id_gastos = toobGastos.id_gastos;
493                 dbo.detalle_gastos.Add(item);
494             }
495
496             dbo.SaveChanges();
497             return true;
498         }
499     }
500
501     #endregion
502 }
503
504     /// <summary>
505     /// Clase solo para obtener detalle de montos por Entidad,unicamente en esta clase se ocupa
506     /// </summary>
507     public class DetalleGastosEntidad {
508         public int id_entidad_x_actividad { get; set; }

```

```
509     public int id_entidad { get; set; }
510     public decimal? subcosto { get; set; }
511     public decimal? iva { get; set; }
512     public decimal? descuento { get; set; }
513     public decimal? total { get; set; }
514 }
515 }
516
517
```

Manual de usuario de sistema de control y fondos y donaciones de proyectos de OIKOS de Nicaragua

Autores

- **Lester Eliezer Huembés López**
- **Juan Carlos Alemán Mayorga**

Tutor: Msc.Ing.Jorge Prado

OIKO de Nicaragua

2016

INDICE

Introducción	2
Propósito del Documento	2
Acceso al Sistema	3
➤ Roles.....	3
➤ Ingresar al Sistema	3
Módulos que Integran el Sistema	6
1. Administración.....	6
a. Entidades	6
b. Cargos de Proyectos.....	9
c. Tipos de Proyectos	11
d. Cuentas Contables.....	13
2. Gestión de Proyectos	15
2.1. Proyectos	15
2.2. Actividades	20
2.3. Donaciones Recibidas.....	26
2.4. Personal en las Actividades de Proyectos	27
2.5. Gatos en Proyectos	28
2.6. Beneficiarios Donaciones	33
3. Personal de las Entidades	35
4. Proveedores.....	38
5. Comprobantes Contables.....	41
6. Gestión de Usuarios del Sistema.....	45

Introducción

El Sistema de Control de Fondos y Donaciones de Proyectos de OIKOS de Nicaragua, fue diseñado con la finalidad de la gestión de proyectos de OIKOS de Nicaragua (aspectos relacionados con el manejo de las actividades, los gastos y las donaciones que se realizan o están contempladas en el plan de ejecución de cada proyecto).

Este sistema ha sido desarrollado para trabajar bajo un ambiente web, con el fin de poder hacer un mejor uso de acceso y de poder permitir la comunicación de la información en todos los medios que sea establecido. Cabe mencionar que existen singularidades de aplicaciones para la gestión de proyectos, pero para el alcance y propósito de este sistema fue desarrollada exclusivamente para OIKOS de Nicaragua.

Propósito del Documento

El propósito de este documento es describir en forma general las necesidades y características más importantes del Sistema de Control de Fondos y Donaciones de Proyectos de OIKOS, enfocándose en como los usuarios pueden ejecutar la diferentes funcionalidades que realiza el sistema.

Este manual irá guiando al usuario en el proceso de registro de proyectos, las actividades del proyecto y sus gastos correspondientes a como también las donaciones que se reciben sean monetarias o de insumos.

Acceso al Sistema

Para ingresar al sistema los usuarios deben contar con una cuenta y contraseña de acceso, previamente creada por el Administrador del Sistema. Cada cuenta de acceso tiene un Rol asignado.

➤ Roles

Un Rol es la función o las funciones que un usuario cumple en el sistema. El sistema cuenta con tres diferentes roles de Usuarios, los cuales manejan diversos permisos de acción y acceso en el sistema:

- **Administrador**, usuarios con este rol tienen permiso a ejecutar todas las acciones y accesos en el sistema (no hay restricciones para estos usuarios).
- **Operador**, los usuarios pueden crear, editar, borrar información cierta información y con restricciones de acceso a las opciones de administración de información de catálogos de información para uso del sistema, a como negación de acceso a gestión de usuarios en el sistema.
- **Consulta**, estos usuarios únicamente tienen permiso de consultar y/o ver la información registradas a como el acceso para los reportes que genera el sistema, este es el nivel más bajo que se maneja en el sistema.

➤ Ingresar al Sistema

1. Para Ingresar al Sistema se debe de abrir un navegador Web de su preferencia (para nuestra presentación trabajaremos con Google Chrome).



Figura 1 Navegador a trabajar

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

2. Se presentará la página para acceder al sistema, la cual nos solicitará un nombre de usuario y contraseña el cual es proporcionado por el administrador del sistema.

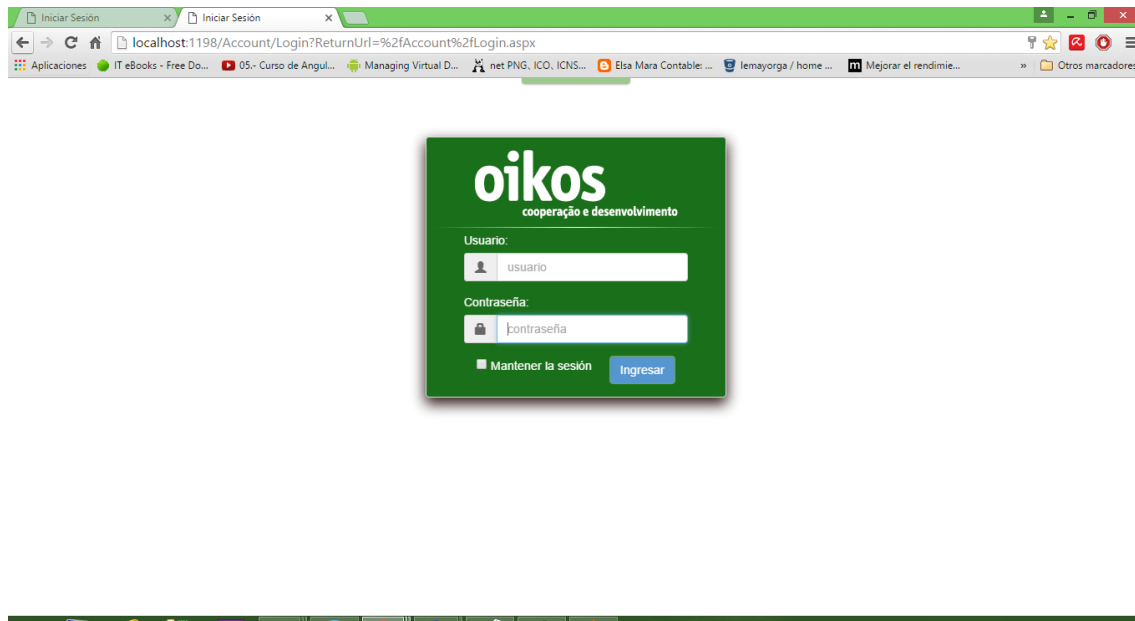


Figura 2 Página para acceder al sistema

- a. El Usuario para acceder al sistema deberá de introducir sus claves de acceso en los campos de Usuario y Contraseña, deben estar previamente dados de alta en el sistema.

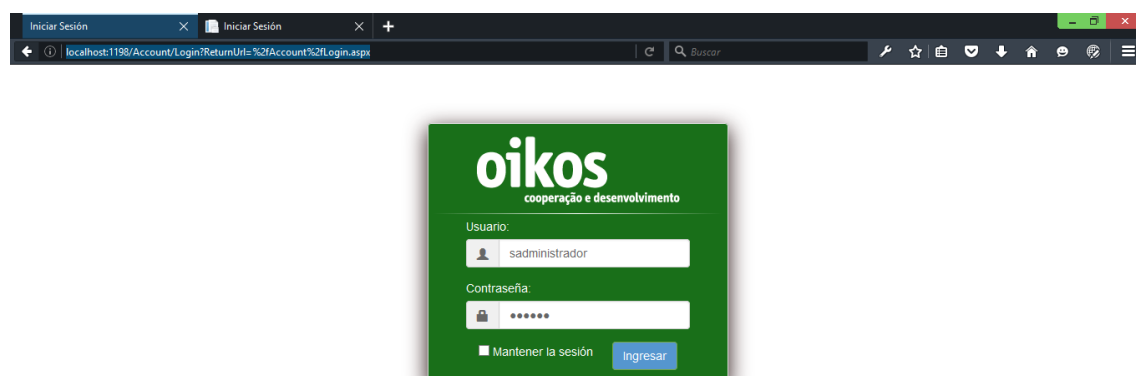


Figura 3 El Usuario para acceder al sistema

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

- b. Si el usuario o la contraseña son incorrectos el sistema mostrará un mensaje indicando que el usuario o la contraseña son incorrectos

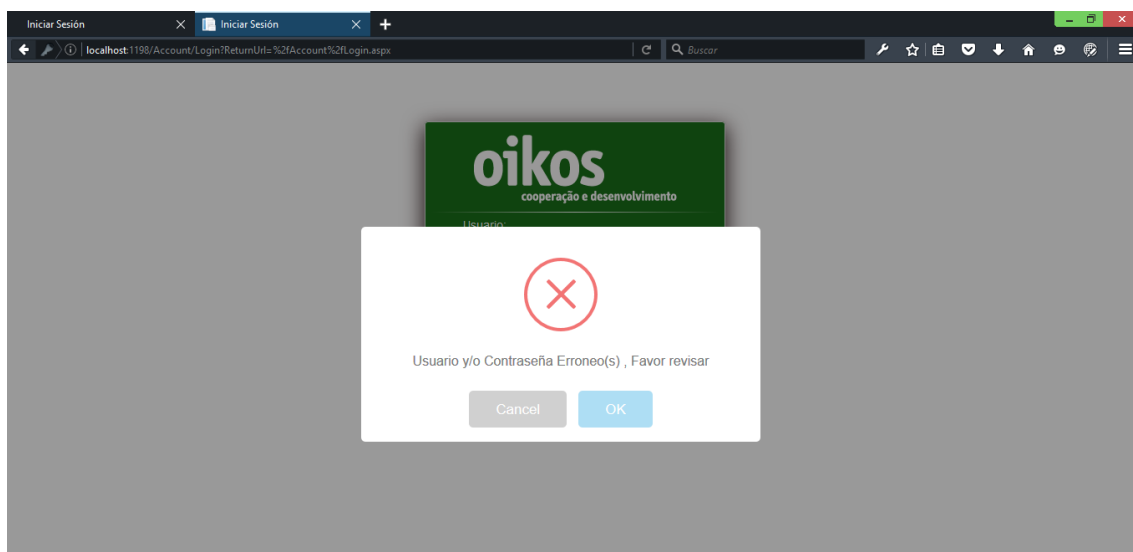


Figura 4 Mensaje indicando que el usuario o la contraseña son incorrectos

3. Una vez que se ha proporcionado un usuario y contraseña correctos el sistema mostrará la pantalla de inicio con las opciones de menú correspondientes al rol del usuario.

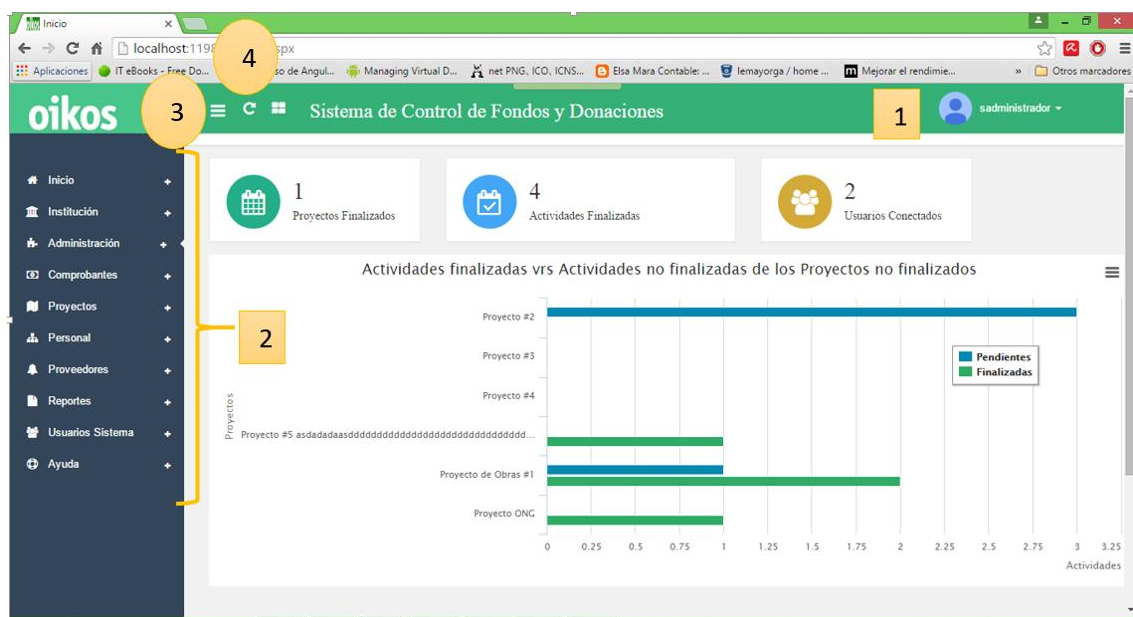


Figura 5 Página Default del Sistema después de loguearse

1. Nombre de Usuario logueado.
2. Opciones del Menú, dependiendo el rol ciertos no se muestran.
3. Botón para ocultar temporalmente el menú lateral.
4. Refrescamiento de página actual.

Módulos que Integran el Sistema

1. Administración

El Módulo de administración cuenta con las siguientes opciones disponibles para los usuarios administradores exclusivamente:

- a. Entidades
- b. Cargos de Proyectos
- c. Tipos de Proyectos
- d. Cuentas Contables

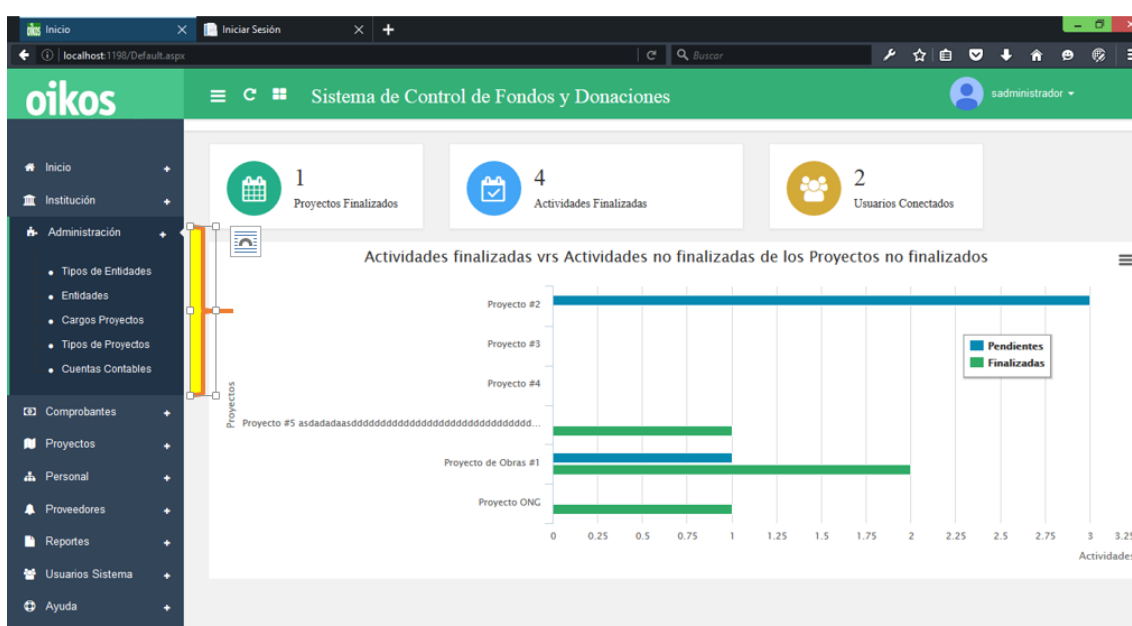


Figura 6

a. Entidades

Se registra las Entidades/ONG que involucran en los distintos proyectos, esta página es la lista de Entidades registradas.

1. Filtros de datos y opciones de exportación de datos.
2. Opción para crear o registrar una nueva Entidad.
3. Opción para seleccionar la Entidad a editar o actualizar información.
4. Opción para Borrar la Entidad seleccionada.

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA



Figura 7

1) Filtros de datos y opciones de exportación de datos.



Figura 8

2) Opción para crear o registrar una nueva Entidad.

Al pulsar esta opción te re direcciona a otro formulario de registro de datos del usuario.

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

Nombre:

Siglas: Nacionalidad:

Dirección:

Correo electrónico: Teléfono: Activo: ☒

Tipo Entidad: Socio: ☐ Donador: ☐

[Seleccione una fila de la tabla de Tipo de Entidades](#)

Mostrar: 10 registros Filtrar datos:

	Nombre	Socio	Donador
<input checked="" type="checkbox"/> Seleccionar	Donador	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> Seleccionar	Donador	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> Seleccionar	Donador	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> Seleccionar	Socio	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> Seleccionar	Socio	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Figura 9

Se llena los campos correspondientes, en caso que no se llena los campos que son requeridos el sistema lo valida mostrando un mensaje de campo requerido.

Nombre:

Siglas: Nacionalidad:

Dirección:

Correo electrónico: Teléfono: Activo: ☒

Tipo Entidad: Socio: ☐ Donador: ☒

[Seleccione una fila de la tabla de Tipo de Entidades](#)

Mostrar: 10 registros Filtrar datos:

	Nombre	Socio	Donador
<input checked="" type="checkbox"/> Seleccionar	Donador	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> Seleccionar	Donador	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Figura 10

Una vez completado los campos se procede a guardar.

- 3) Opción para seleccionar la Entidad a editar o actualizar información.
Esta opción es semejante a la registrar un nuevo registro de Entidad, se seleccón la entidad y te re direcciona a la página de creación, una vez actualizados se guardan los datos.
- 4) Opción para Borrar la Entidad seleccionada.
Se validad si la entidad no está enlazada con alguna otra información, en caso de ser así, se tendría que borrar los enlaces relacionados a la entidad.

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

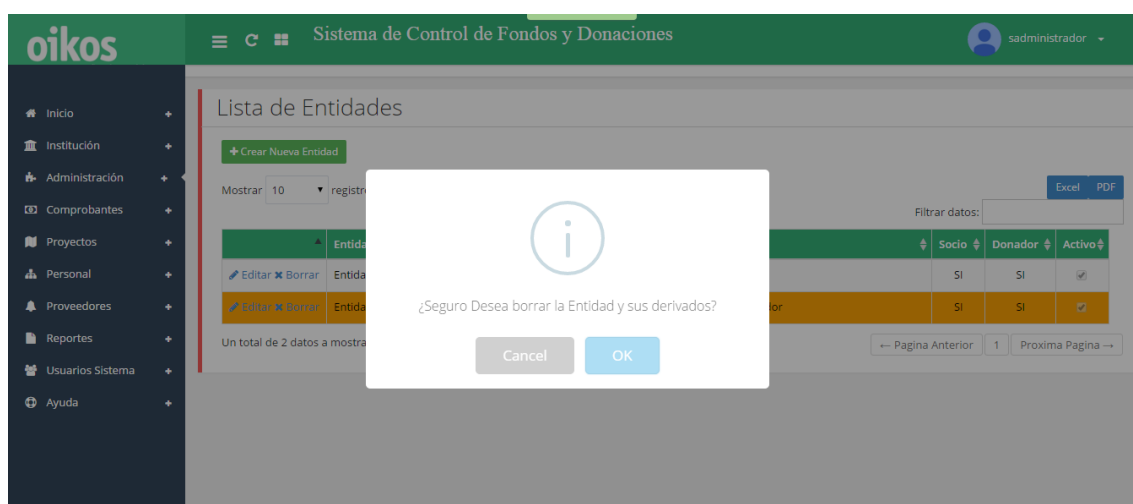


Figura 11

b. Cargos de Proyectos

Se registra los cargos de proyectos que involucran en los distintos personal en los proyectos, esta página es la lista catálogo de cargos en proyectos de personal registradas.

1. Filtros de datos y opciones de exportación de datos.
2. Opción para crear o registrar un nuevo Cargo.
3. Opción para seleccionar el Cargo a editar o actualizar información.
4. Opción para Borrar un Cargo seleccionada.



Figura 12

- 1) Filtros de datos y opciones de exportación de datos.

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

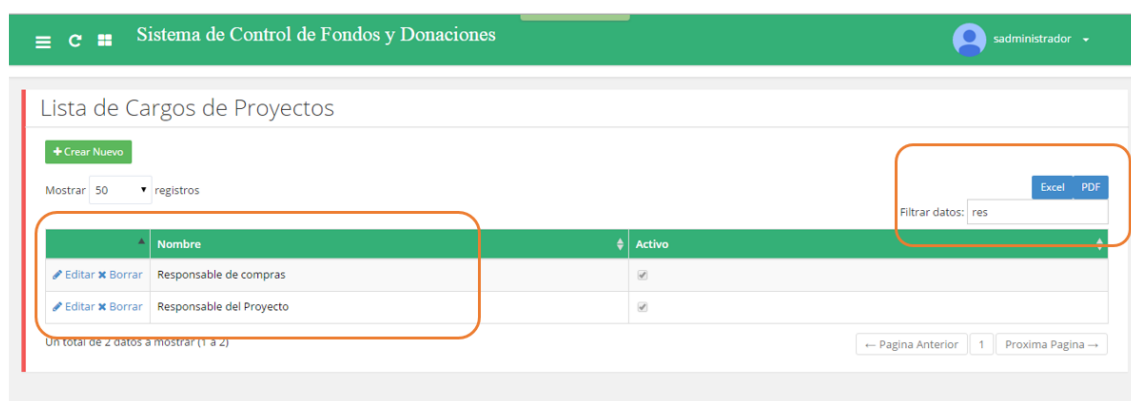


Figura 13

- 2) Opción para crear o registrar un nuevo Cargo.
Al pulsar la opción de crear nuevo, este abre un modal o ventana emergente para registrar un nuevo cargo de proyecto.

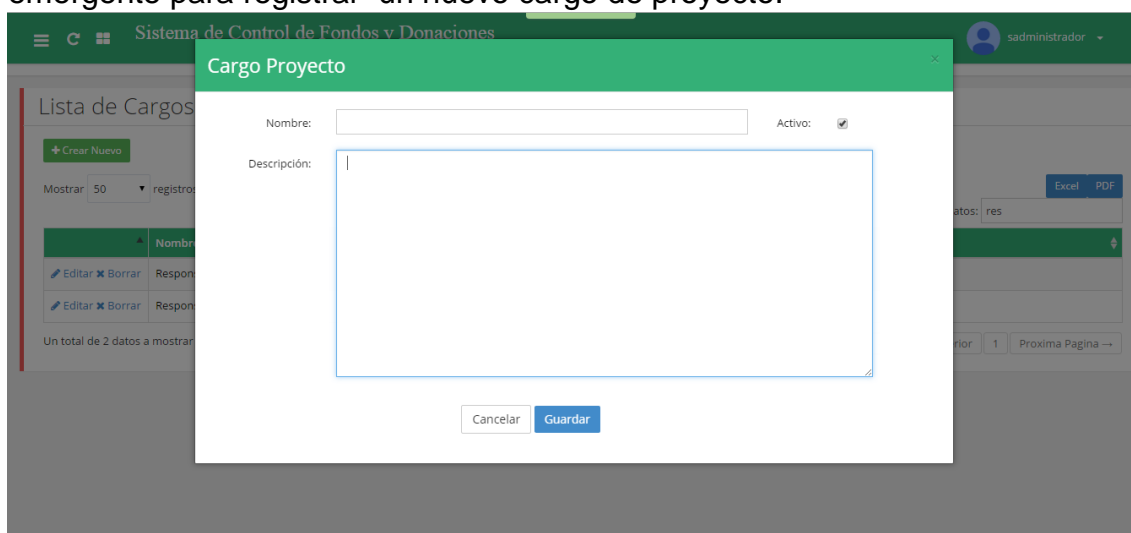


Figura 14

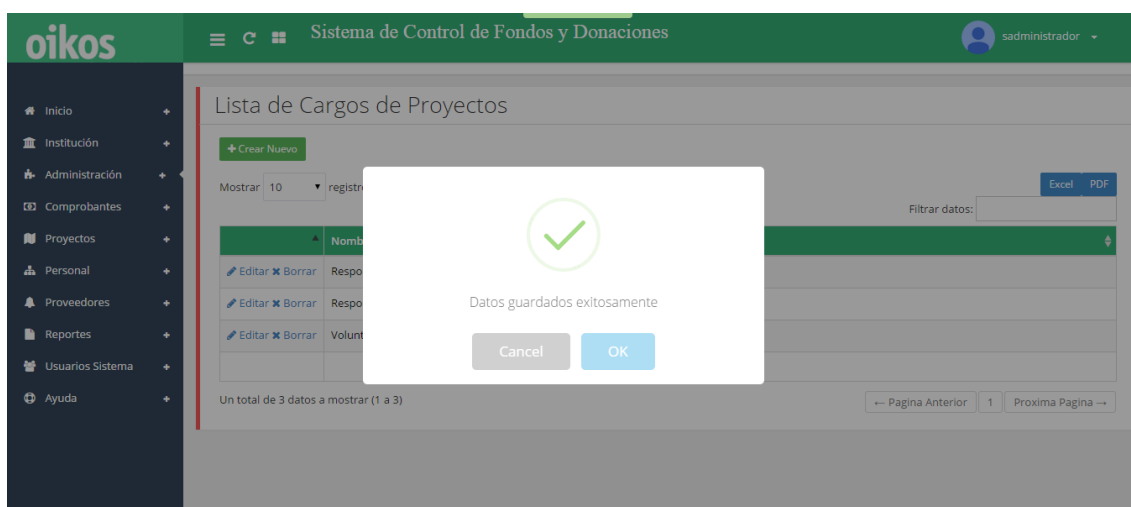


Figura 15

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

- 3) Opción para seleccionar el Cargo a editar o actualizar información.
Esta opción es semejante a la registrar un nuevo registro de cargo para el personal en los proyectos, se selecció el cargo y se abre la venta modal o ventana emergente.
- 4) Opción para Borrar un Cargo seleccionada.
Se validad si el cargo no está enlazada con alguna otra información de personal de proyecto, en caso de ser así, se tendría que borrar los enlaces relacionados al cargo.

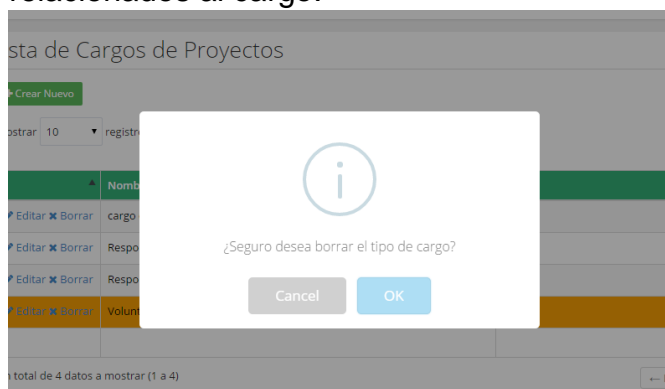


Figura 16

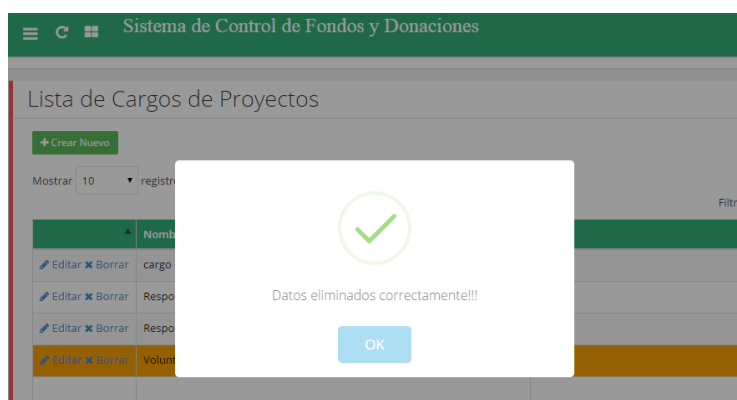


Figura 17

c. Tipos de Proyectos

Se registra las diferentes categorías de tipos de proyectos que maneja la ONG (catálogo de tipos de proyectos).

1. Filtros de datos y opciones de exportación de datos.
2. Opción para crear o registrar un nuevo tipo de proyecto.
3. Opción para seleccionar el Tipo de Proyecto a editar o actualizar información.
4. Opción para Borrar un Tipo de Proyecto seleccionada.

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

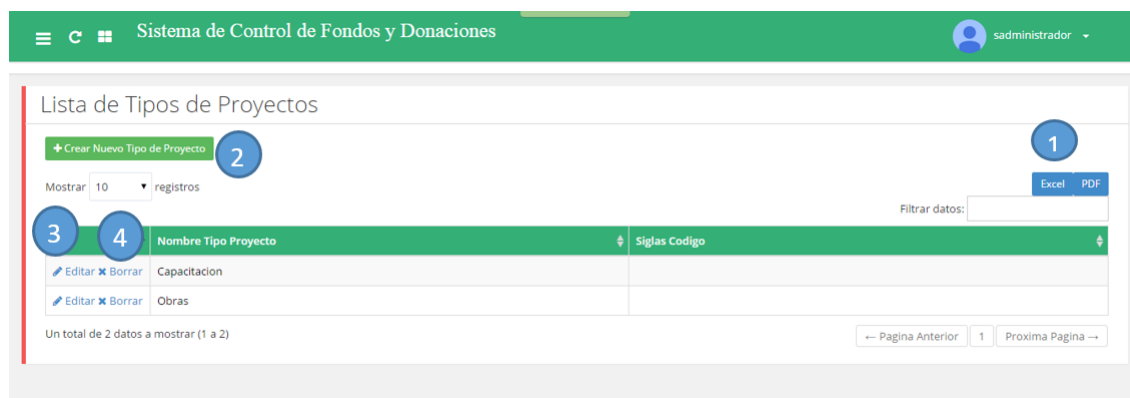


Figura 18

- 1) Filtros de datos y opciones de exportación de datos.

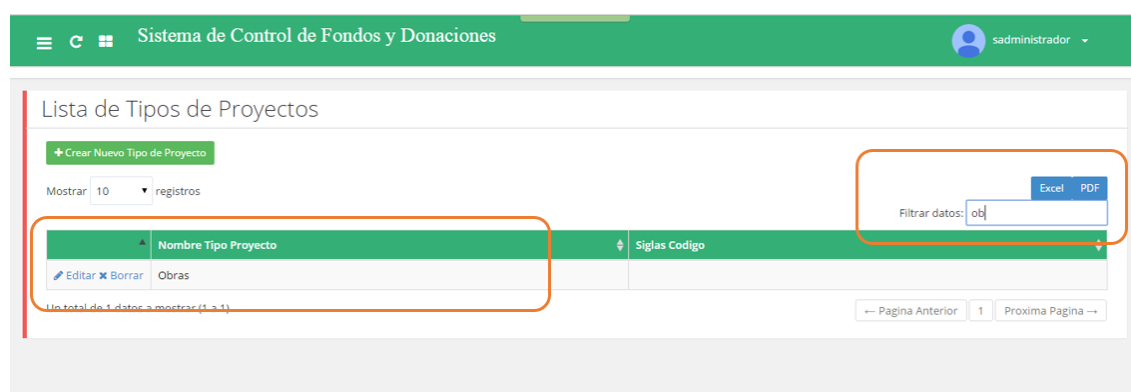


Figura 19

- 2) Opción para crear o registrar un nuevo tipo de proyecto.
Al pulsar la opción de crear nuevo, este abre un modal o ventana Emergente para registrar un nuevo tipo de proyecto para el catalogo.

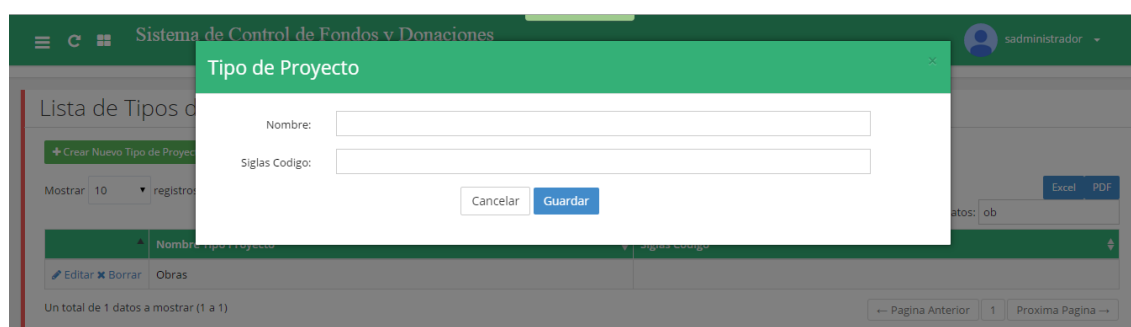


Figura 20

- 3) Opción para seleccionar el Tipo de Proyecto a editar o actualizar información.

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

Esta opción es semejante a la registrar un nuevo registro de Tipo de Proyecto para el personal en los proyectos, se selección el Tipo de Proyecto y se abre la venta modal o ventana emergente.

- 4) Opción para Borrar un Tipo de Proyecto seleccionada.
Se valida si el tipo de proyecto no está enlazada con alguna otra información de personal de proyecto, en caso de ser así, se tendría que borrar los enlaces relacionados al tipo de proyecto.

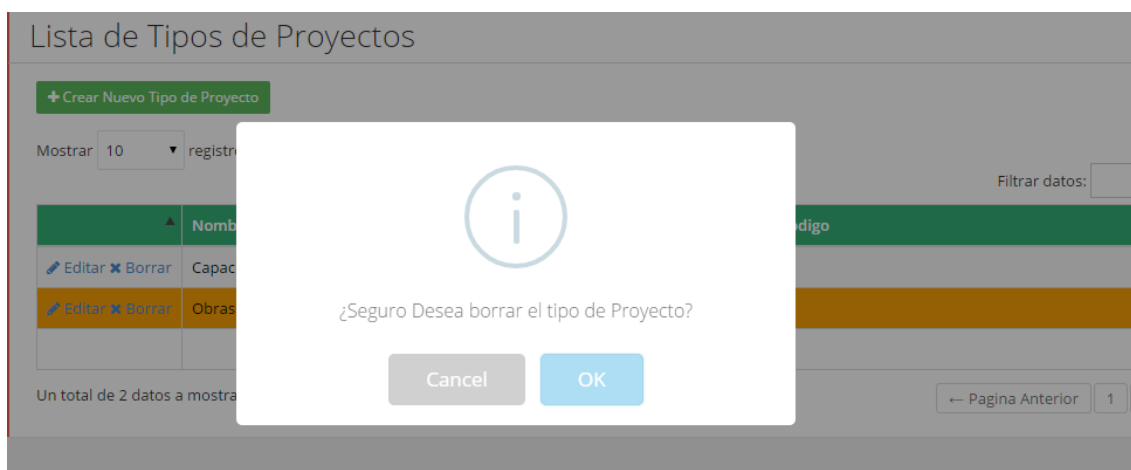


Figura 21

d. Cuentas Contables

Se registra las diferentes cuentas contables con las cuales trabaja la ONG, para fin de registrar un catálogo en el sistema, para ser usadas a la hora de crear comprobantes de los gastos realizados en los proyectos.

1. Filtros de datos y opciones de exportación de datos.
2. Opción para crear o registrar una nueva cuenta contable.
3. Opción para seleccionar la cuenta contable a editar o actualizar información.



Figura 22

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

1) Filtros de datos y opciones de exportación de datos.



Figura 23

2) Opción para crear o registrar una nueva cuenta contable. Al pulsar la opción este abre un modal o ventana emergente para registrar una nueva cuenta contables para el catálogo.

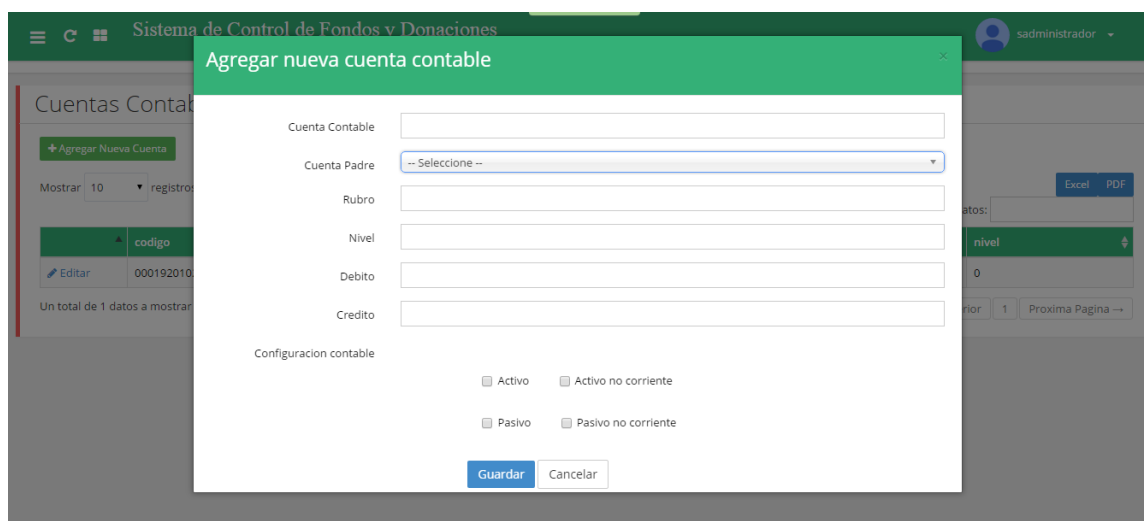


Figura 24

3) Opción para seleccionar la cuenta contable a editar o actualizar información. Esta opción es semejante a la registrar una nueva cuenta contables, se selección la cuenta contable y se abre la venta modal o ventana emergente.

2. Gestión de Proyectos

2.1. Proyectos

La gestión de proyectos maneja la parte de catálogo de proyectos, para llevar su histórico y actual seguimiento a los mismos, las acciones que se pueden realizar son las siguientes:



Figura 25

En la lista de Proyectos se enumeran las siguientes acciones que se pueden realizar:

1. Filtros de datos y opciones de exportación de datos.
2. Opción para crear o registrar un nuevo proyecto.
3. Opción para seleccionar un proyecto a editar o actualizar información.
4. Re direccionamientos a ejecuciones del Proyecto
 - 4.1 Lista de Actividades del Proyecto y Gastos de Actividades
 - 4.2 Lista de Gastos de Actividades
5. Opción para borrar un Proyecto

The screenshot shows the '4.2 de Proyectos' page. It includes a header with navigation links: 'Lista de Actividades', 'Lista de Gastos-Actividades', and 'Crear Nuevo proyecto'. Below the header is a table with columns: 'Codigo', 'Nombre', and 'Estado'. The table lists five projects. Annotations are present: '4.2' for the page title, '2' for the 'Crear Nuevo proyecto' link, '1' for the 'Excel' and 'PDF' export buttons, '3' for the 'Editar' button, '4.1' for the 'Actividades' button, and '5' for the 'Borrar' button. The table data is as follows:

	Codigo	Nombre	Estado
Editar Actividades Borrar	p-0000000000023	Proyecto ONG	
Editar Actividades Borrar	p-0000000000024	Proyecto de Obras #1	
Editar Actividades Borrar	p-0000000001024	Proyecto #2	
Editar Actividades Borrar	p-0000000001025	Proyecto #3	
Editar Actividades Borrar	p-0000000001026	Proyecto #4	

Un total de 5 datos a mostrar (1 a 5)

Figura 26

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

1) Filtros de datos y opciones de exportación de datos.

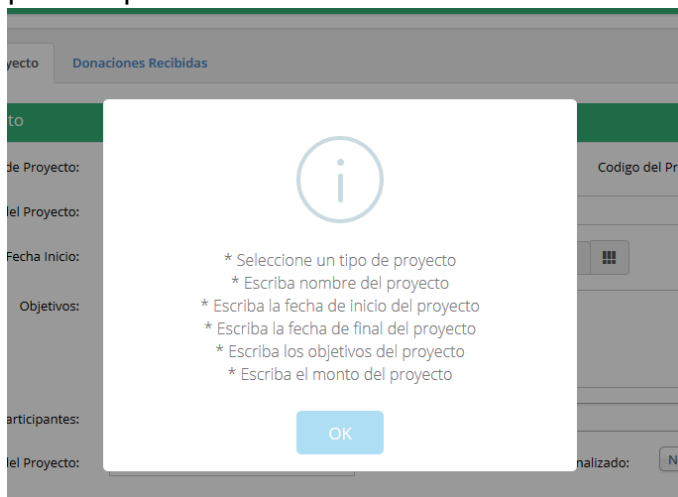


2) Opción para crear o registrar un nuevo proyecto.

Al pulsar la opción de crear este re direcciona a otra página para poder registrar la información correspondiente al crear un proyecto con sus datos relevantes.



A como en otros formularios, se validan los datos que son requeridos para su posterior almacenamiento o creación del proyecto.



MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

Sistema de Control de Fondos y Donaciones

administrador

Datos del Proyecto Donaciones Recibidas 10

Proyecto

1 Tipo de Proyecto: Capacitacion 2 Codigo del Proyecto:

3 Nombre del Proyecto: Proyecto de Prueba # 6

4 Fecha Inicio: 01/01/2016 5 Fecha Final: 21/04/2016

6 Objetivos: Objetivos del Proyecto

7 Entidades Participantes: Entidad 1 X

8 Monto del Proyecto: 36,800.00 9 Proyecto Finalizado: NO

Lista de Proyectos Guardar Proyecto

1. Tipo de Proyecto
2. Código del Proyecto (Generado automáticamente al guardar)
3. Nombre del Proyecto
4. Fecha de Inicio del Proyecto
5. Fecha de finalización del Proyecto
6. Objetivos del Proyecto
7. Entidades o ONG participantes (en caso de que existan involucrados)
8. Monto o Presupuesto del Proyecto
9. Estado del Proyecto(Únicamente 2 estados, inicio y finalizado)
10. TAB de ingreso de Datos de Donaciones del Proyecto(guardar primero los datos del proyecto)
11. Entidad donante
12. Insumo que se recibe de donación
13. Cantidad del Insumo
14. Fecha de la Donación
15. Comentarios (opcional)
16. Agregar a la lista o tabla de donaciones de abajo(ojo no se guarda en lo absoluto)
17. Guardar datos nuevos o actualizados de las donaciones
18. TAB de datos del proyecto

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

Sistema de Control de Fondos y Donaciones

administrador

Datos del Proyecto Donaciones Recibidas

Donacion

11 Entidad: *** Seleccione ***

12 Nombre articulo:

13 Cantidad:

14 Fecha:

15 Comentarios:

Lista de Proyectos Limpiar Agregar Elemento 16 Guardar Donaciones Registradas 17

Nombre	Articulo	Cantidad	Fechas	Observacion
No hay datos en la tabla				

Si el proyecto se finaliza, se finalizada todo lo relacionado al el, con opción de ver los datos almacenados.

3) Opción para seleccionar un proyecto a editar o actualizar información.

Al pulsar la opción de editar, este redirección a la página como la de crear un proyecto, con diferencia de que carga los datos correspondiente al proyecto, listo para editar actualizar.

Sistema de Control de Fondos y Donaciones

administrador

Datos del Proyecto Donaciones Recibidas

Proyecto

Tipo de Proyecto: Obras Codigo del Proyecto: p-00000000000024

Nombre del Proyecto: Proyecto de Obras #1

Fecha Inicio: 01/09/2015 Fecha Final: 31/12/2015

Objetivos: asdasdasd

Entidades Participantes: Entidad 1

Monto del Proyecto: 5,000.00 Proyecto Finalizado: NO

Lista de Proyectos Editar

4) Re direccionamientos a ejecuciones del Proyecto

- Lista de Actividades del Proyecto y Gastos de Actividades
- Lista de Gastos de Actividades

El uso de estos se explica posterior al manual.

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

The screenshot shows the 'Lista de Proyectos' page. Callout A points to the 'Mostrar 10 registros' dropdown, and callout B points to the 'Crear Nuevo proyecto' button. The table lists five projects with their codes and names. The 'Borrar' link for the first project is highlighted.

	Codigo	Nombre	Estado
Editar Actividades Borrar	p-00000000000023	Proyecto ONG	
Editar Actividades Borrar	p-00000000000024	Proyecto de Obras #1	
Editar Actividades Borrar	p-000000000001024	Proyecto #2	
Editar Actividades Borrar	p-000000000001025	Proyecto #3	
Editar Actividades Borrar	p-000000000001026	Proyecto #4	

5) Opción para borrar un Proyecto

Se valida si el proyecto no está enlazada con alguna otra información (actividades, gastos), en caso de ser así, se tendría que borrar los enlaces relacionados al proyecto.

The screenshot shows the 'Lista de Proyectos' page. The 'Borrar' link for the first project is highlighted with a red box. The table lists five projects with their codes and names. The 'Borrar' link for the first project is highlighted.

	Codigo	Nombre	Estado
Editar Actividades Borrar	p-00000000000023	Proyecto ONG	
Editar Actividades Borrar	p-00000000000024	Proyecto de Obras #1	
Editar Actividades Borrar	p-000000000001024	Proyecto #2	
Editar Actividades Borrar	p-000000000001025	Proyecto #3	
Editar Actividades Borrar	p-000000000001026	Proyecto #4	

The screenshot shows a confirmation dialog box with the text '¿Seguro Desea borrar el proyecto?' and two buttons: 'Cancel' and 'OK'.

2.2. Actividades

La gestión de actividades corresponde a las actividades que se realizan en el proyecto, actividades de todo tipo, reuniones, entrega de dinero, a como donaciones de dinero, capacitaciones, entrega de materiales, etc., las acciones que se pueden realizar son las siguientes:

Un total de 6 datos a mostrar (1 a 6)

La última columna define o muestra el estado de la actividad, cuando está finalizada se muestra de color verde y al estar en este estado, los datos almacenados a ella y los relacionados a ella, únicamente se puede ver o visualizar, no editar.

1. Filtros de datos por proyecto.
2. Re direccionamientos a ejecuciones de la Actividades
 - a. Lista de Proyectos
 - b. Lista de Gastos de Actividades
3. Filtros de datos y opciones de exportación de datos.
4. Crear una nueva actividad.
5. Editar datos de una actividad seleccionada.
6. Opción para borrar una actividad seleccionada.

- 1) Filtros de datos por proyecto.

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

Sistema de Control de Fondos y Donaciones administrador

Actividades de los Proyectos

Filtro por Proyecto(s): Proyecto #2

Lista de Proyectos Lista de Gastos-Actividades + Crear Nueva Actividad

Mostrar 10 registros

Excel PDF

	proyecto	Actividad	Codigo Actividad	Actividad Padre	Estado
Editar Gastos Borrar	Proyecto #2	Compra de libros	a-0000000001020		
Editar Gastos Borrar	Proyecto #2	COMRA DE PIZARRA	a-0000000001021		

Un total de 2 datos a mostrar (1 a 2)

← Pagina Anterior 1 Proxima Pagina →

- 2) Re direccionamientos a ejecuciones de la Actividades
 - a. Lista de Proyectos (re direcciona)
 - b. Lista de Gastos de Actividades (Re direcciona)

Sistema de Control de Fondos y Donaciones administrador

Actividades de los Proyectos

Filtro por Proyecto(s): *** Seleccione ***

Lista de Proyectos Lista de Gastos-Actividades + Crear Nueva Actividad

Mostrar 10 registros

Excel PDF

	proyecto	Actividad	Codigo Actividad	Actividad Padre	Estado
Editar Gastos Borrar	Proyecto de Obras #1	dasddasdata	a-00000000000022	P#1 A#2fsdfsfsfd	
Editar Gastos Borrar	Proyecto #2	Compra de libros	a-0000000001020		
Editar Gastos Borrar	Proyecto #2	COMRA DE PIZARRA	a-0000000001021		
Ver Gastos	Proyecto ONG	P#1 A#2	a-00000000000019		Finalizada
Ver Gastos	Proyecto de Obras #1	P#1 A#2fsdfsfsfd	a-00000000000020		Finalizada
Ver Gastos	Proyecto de Obras #1	P#1 A#2fsdfsfsfd	a-00000000000021		Finalizada

Un total de 6 datos a mostrar (1 a 6)

← Pagina Anterior 1 Proxima Pagina →

- 3) Filtros de datos y opciones de exportación de datos.

Sistema de Control de Fondos y Donaciones administrador

Actividades de los Proyectos

Filtro por Proyecto(s): *** Seleccione ***

Lista de Proyectos Lista de Gastos-Actividades + Crear Nueva Actividad

Mostrar 10 registros

Excel PDF

	proyecto	Actividad	Codigo Actividad	Actividad Padre	Estado
Editar Gastos Borrar	Proyecto de Obras #1	dasddasdata	a-00000000000022	P#1 A#2fsdfsfsfd	
Editar Gastos Borrar	Proyecto #2	Compra de libros	a-0000000001020		
Editar Gastos Borrar	Proyecto #2	COMRA DE PIZARRA	a-0000000001021		
Ver Gastos	Proyecto ONG	P#1 A#2	a-00000000000019		Finalizada
Ver Gastos	Proyecto de Obras #1	P#1 A#2fsdfsfsfd	a-00000000000020		Finalizada
Ver Gastos	Proyecto de Obras #1	P#1 A#2fsdfsfsfd	a-00000000000021		Finalizada

Un total de 6 datos a mostrar (1 a 6)

← Pagina Anterior 1 Proxima Pagina →

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

4) Crear una nueva actividad.

Al pulsar la opción de crear este re direcciona a otra página para poder registrar la información correspondiente a la actividad y sus datos

Sistema de Control de Fondos y Donaciones

administrador

Actividades de los Proyectos

Filtro por Proyecto(s): *** Seleccione ***

Lista de Proyectos Lista de Gastos-Actividades **Crear Nueva Actividad**

Mostrar 10 registros

Excel PDF

Filtrar datos:

	proyecto	Actividad	Codigo Actividad	Actividad Padre	Estado
Editar Gastos Borrar	Proyecto de Obras #1	dasddasddada	a-00000000000022	P#1 A#2fsdfsfsfd	
Editar Gastos Borrar	Proyecto #2	Compra de libros	a-00000000001020		
Editar Gastos Borrar	Proyecto #2	COMRA DE PIZARRA	a-00000000001021		
Ver Gastos	Proyecto ONG	P#1 A#2	a-00000000000019		Finalizada
Ver Gastos	Proyecto de Obras #1	P#1 A#2fsdfsfsfd	a-00000000000020		Finalizada
Ver Gastos	Proyecto de Obras #1	P#1 A#2fsdfsfsfd	a-00000000000021		Finalizada

Un total de 6 datos a mostrar (1 a 6)

← Pagina Anterior 1 Proxima Pagina →

A como en otros formularios, se validan los datos que son requeridos para su posterior almacenamiento o creación de la actividad.

!

- * Seleccione el proyecto
- * Escriba la descripción de la actividad
- * Escriba la fecha de inicio de la actividad
- * Escriba la fecha de finalización de la actividad
- * Agregue al menos una ONG/Entidad a la lista

OK

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

1. Descripción o nombre de la actividad
2. Proyecto al cual pertenece la actividad
3. Fecha de Inicio o en que se realizó su inicio
4. Fecha de Finalización de la actividad
5. Actividad padre (opcional o si lo amerita)
6. Estado de la actividad, (si se finaliza la información será únicamente visible para posterior acciones)
7. Opción para agregar entidades participantes en la entidad
8. Campo de Observaciones o nota (opcional)
9. Botón de guardar datos de la actividad
10. TAB de datos de personal de entidades en la actividad

11. TAB de datos de la actividad
12. Opción para agregar personal entidad a la actividad

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

13. Formulario para agregar personal a la actividad, el proyecto se establece automático según el proyecto seleccionado al crear la actividad.
14. Se selecciona la entidad participante entre las entidades agregadas a la actividad
15. El cargo que rige la persona agregada.

Nota: antes de agregar personal debe guardar los datos de la actividad.

Entidad/ONG	Personal	Cargo
OIKO - Es Entrada: NO	Ronaldo Tinoco Centeno	Responsable del Proyecto

5) Editar datos de una actividad seleccionada.

Al pulsar la opción de editar, este redirección a la página como la de crear una actividad, con diferencia de que carga los datos correspondiente a la actividad, listo para editar actualizar.

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

Sistema de Control de Fondos y Donaciones | administrador

Datos de la Actividad | **Personal en Actividad**

Actividad Realizada - Código Actividad: a-0000000000022

Actividad(Descripción):

Proyecto:

Fecha de Inicio: Fecha de Finalización:

Actividad Padre:

Entidades Participantes: [+ Agregar Nueva Entidad](#) | Actividad Finalizada:

Entidad	Monto	Es Entrad	Observaciones
OIKO	33.00	NO	

Observaciones:

[Retornar](#) [Editar](#)

6) Opción para borrar una actividad seleccionada.

Se valida si la actividad no está enlazada con alguna otra información (gatos), en caso de ser así, se tendría que borrar los enlaces relacionados a la actividad.

Sistema de Control de Fondos y Donaciones | administrador

Actividades de los Proyectos

Filtro por Proyecto(s):

[Lista de Proyectos](#) [Lista de Gastos-Actividades](#) [+ Crear Nueva Actividad](#)

Mostrar registros

	proyecto	Actividad	Código Actividad	Actividad Padre	Estado
Editar Gastos Borrar	Proyecto de Obras #1	dasddasda	a-0000000000022	P#1 A#2fsdfsfsd	Finalizada
Editar Gastos Borrar	Proyecto #2	Compra de libros	a-0000000001020		
Editar Gastos Borrar	Proyecto #2	COMRA DE PIZARRA	a-0000000001021		
Ver Gastos	Proyecto ONG	P#1 A#2	a-0000000000019		Finalizada
Ver Gastos	Proyecto de Obras #1	P#1 A#2fsdfsfsd	a-0000000000020		Finalizada
Ver Gastos	Proyecto de Obras #1	P#1 A#2fsdfsfsd	a-0000000000021		Finalizada

Un total de 6 datos a mostrar (1 a 6)

[← Pagina Anterior](#) [1](#) [Proxima Pagina →](#)

Las actividades finalizadas no pueden ser borradas, como se observa en la figura, no traen la opción de borrar.

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

2.3. Donaciones Recibidas

Para acceder a la opción de las donaciones recibidas en el proyecto, se pueden encontrar en la pestaña contigua a datos del proyecto.

The screenshot shows the 'Sistema de Control de Fondos y Donaciones' interface. The left sidebar contains a menu with options: Inicio, Institución, Administración, Reportes, Crear Comprobante, Proyectos, Beneficiarios, Personal, Proveedores, Usuarios Sistema, and Ayuda. The main area is titled 'Datos del Proyecto' and has a tab labeled 'Donaciones Recibidas' highlighted with a red box. A blue oval points to this tab with the text 'Pestaña de datos de Donaciones recibidas del Proyecto'. Below the tab, there is a form for 'Proyecto' with fields for 'Tipo de Proyecto' (dropdown), 'Codigo del Proyecto' (text), 'Nombre del Proyecto' (text), 'Fecha Inicio' and 'Fecha Final' (calendars), 'Objetivos' (text area), 'Entidades Participantes' (dropdown), 'Monto del Proyecto' (text), and 'Proyecto Finalizado' (dropdown). At the bottom are buttons for 'Lista de Proyectos' and 'Guardar Proyecto'.

The screenshot shows the 'Donacion' form. It has fields for 'Entidad' (dropdown, labeled 1), 'Nombre articulo' (text, labeled 2), 'Cantidad' (text, labeled 3), 'Fecha' (calendar, labeled 4), and 'Comentarios' (text area, labeled 5). Below these are buttons for 'Lista de Proyectos', 'Limpiar', and 'Agregar Elemento' (labeled 6). A blue oval labeled 7 points to the 'Agregar Elemento' button. At the bottom right is a button labeled 'Guardar Donaciones Registradas' (labeled 8). Below the form is a table with columns: Nombre, Articulo, Cantidad, Fechas, and Observacion. The table is currently empty, showing 'No hay datos en la tabla'.

1. Nombre de la Entidad participa en el proyecto la cual realiza la donación.
2. Nombre el rubro (ítem) de donación.
3. Cantidad del recurso recibido.
4. Fecha de la donación.
5. Comentario (dato opcional).
6. Agregar al ítem al grid o lista de donaciones (aún no se guarda la información).
7. Procesar los datos agregados al grid para su posterior almacenamiento.

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

The screenshot shows the 'Donaciones Recibidas' (Received Donations) module. The interface includes a sidebar with navigation options like 'Inicio', 'Institución', 'Administración', 'Reportes', 'Crear Comprobante', 'Proyectos', 'Beneficiarios', 'Personal', 'Proveedores', 'Usuarios Sistema', and 'Ayuda'. The main content area has tabs for 'Datos del Proyecto' and 'Donaciones Recibidas'. Under 'Donaciones Recibidas', there is a 'Donacion' form with fields for 'Entidad' (a dropdown menu), 'Nombre articulo', 'Cantidad', 'Fecha', and 'Comentarios'. Below the form are buttons for 'Lista de Proyectos', 'Limpiar', 'Agregar Elemento', and 'Guardar Donaciones Registradas'. A table below the form displays a list of donations with columns: 'Nombre', 'Articulo', 'Cantidad', 'Fechas', and 'Observacion'. The table contains one entry: 'Care Nicaragua', 'combustible para vehiculo d...', '30.00', '08/02/2016', and 'unida en galones'.

2.4. Personal en las Actividades de Proyectos

Módulo de personal que participa en las actividades de los proyectos, personal de las entidades participantes. Esta opción la encontramos en una pestaña continua a la de datos de la actividad, al momento de editar una actividad.

The screenshot shows the 'Personal en la Actividad' (Personal in the Activity) module. The interface includes a sidebar with navigation options. The main content area has tabs for 'Datos de la Actividad' and 'Personal en la Actividad'. Under 'Personal en la Actividad', there is a form with fields for 'Entidad/ONG', 'Personal', and 'Cargo'. Below the form are buttons for 'Agregar un Personal a la Actividad', 'Retornar', 'Excel', and 'PDF'. A table below the form displays a list of personnel with columns: 'Entidad/ONG', 'Personal', and 'Cargo'. The table contains one entry: 'Care Nicaragua', 'combustible para vehiculo d...', '30.00', '08/02/2016', and 'unida en galones'.

1) Agregar un Personal a la actividad

Se abre un modal para capturar la información.

The screenshot shows the 'Personal en Actividad del Proyecto' modal. The modal has a title bar and a close button. It contains a form with fields for 'Proyecto', 'Entidad/ONG', 'Persona Participante', and 'Cargo de Persona'. Below the form are buttons for 'Cancelar' and 'Agregar'. The modal is numbered 1 through 5, indicating the steps to add personnel to an activity.

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

1. Proyecto al que se referencia.
2. Entidad de origen de la personal participante.
3. Persona participante.
4. Cargo de la persona
5. Guardar los datos a la actividad.

The screenshot shows the 'Sistema de Control de Fondos y Donaciones' interface. The user is logged in as 'sadministrador'. The 'Personal en la Actividad' section is active, showing a table with one record. The record details are: Entidad/ONG: OIKO - Es Entrada: NO, Personal: Jorger Zepeda, Cargo: Coordinador. The interface includes buttons for 'Agregar un Personal a Actividad', 'Retornar', 'Excel', 'PDF', and 'Filtrar datos'. The table has columns for 'Entidad/ONG', 'Personal', and 'Cargo'. The status 'Un total de 1 datos a mostrar (1 a 1)' is displayed at the bottom.

Entidad/ONG	Personal	Cargo
OIKO - Es Entrada: NO	Jorger Zepeda	Coordinador

2.5. Gatos en Proyectos

La gestión de gastos corresponde a las compras y/o gastos realizados durante el transcurso del proyecto en cada una de las actividades que se realicen, la mayoría de las actividades tienen un gato asociado, que luego este tendrá que ser agregado a un comprobante contable. Para esta gestión hay 3 momentos de estados, por así decir, uno cuando la actividad que lo genera no está finalizada, cuando esta así el gasto se puede editar, dos cuando la actividad está finalizada, el gasto no se puede editar y tres cuando el gasto se ha agregado al comprobante de diario, en este sucede como si la actividad que lo precede estuviera finalizada, las acciones que se pueden realizar son las siguientes:

The screenshot shows the 'Gastos por Actividades-Proyectos' interface. The user is logged in as 'sadministrador'. The interface includes dropdowns for 'Proyecto' and 'Actividad'. The 'Lista de Proyectos' and 'Lista de Actividades' tabs are visible. The 'Crear Nueva Gasto' button is highlighted. The table shows one record with the following details: Proyecto: Proyecto ONG, Actividad: P#1 A#2, Codigo Gasto: g-0000000000014, Factura: 3434, Fecha: 26/10/2015, Proveedor: Proveedor Electrico. The interface includes buttons for 'Excel', 'PDF', 'Filtrar datos', and 'Ver'. The status 'Un total de 1 datos a mostrar (1 a 1)' is displayed at the bottom. Numbered annotations (1-6) highlight specific elements: 1. Proyecto dropdown, 2. Actividad dropdown, 3. Crear Nueva Gasto button, 4.1. Ver button, 4.2. Filter button, 5. Tab bar, 6. Status text.

Proyecto	Actividad	Codigo Gasto	Factura	Recibo	Fecha	Proveedor
Proyecto ONG	P#1 A#2	g-0000000000014	3434		26/10/2015	Proveedor Electrico

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

1. Filtros de datos por proyecto.
2. Filtros de datos por actividad.
3. Filtros de datos y opciones de exportación de datos.
4. Re direccionamientos a ejecuciones de la Actividades
 - a. Lista de Proyectos
 - b. Lista de Gastos de Actividades
5. Crear un nuevo Gasto-Actividad.
6. Editar datos de un gasto seleccionado.
7. Opción para borrar un gasto seleccionado.

1) Filtros de datos por proyecto.

The screenshot shows the 'Gastos por Actividades-Proyectos' interface. At the top, there's a green header with the system name and a user profile. Below the header, the title 'Gastos por Actividades-Proyectos' is displayed. There are two dropdown menus: 'Proyecto' (set to 'Proyecto ONG') and 'Actividad' (set to '*** Seleccione ***'). Below these are three buttons: 'Lista de Proyectos', 'Lista de Actividades', and 'Crear Nueva Gasto'. A 'Mostrar' dropdown is set to '10 registros'. To the right are 'Excel' and 'PDF' export buttons. A table with one row of data is shown, with columns: Proyecto, Actividad, Código Gasto, Factura, Recibo, Fecha, and Proveedor. The data row shows 'Proyecto ONG', 'P#1 A#2', 'g-00000000000014', '3434', an empty 'Recibo' field, '26/10/2015', and 'Proveedor Electrico'. At the bottom, it says 'Un total de 1 datos a mostrar (1 a 1)' and has pagination controls for 'Página Anterior', '1', and 'Próxima Página'.

2) Filtros de datos por actividad.

This screenshot is identical to the previous one, but the 'Proyecto' dropdown is now set to '*** Seleccione ***' and the 'Actividad' dropdown is also set to '*** Seleccione ***'. The table still shows the same data row for 'Proyecto ONG' and 'P#1 A#2'.

3) Filtros de datos y opciones de exportación de datos.

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

Sistema de Control de Fondos y Donaciones

administrador

Gastos por Actividades-Proyectos

Proyecto: *** Seleccione ***

Actividad: *** Seleccione ***

Lista de Proyectos | Lista de Actividades | **+ Crear Nueva Gasto**

Mostrar 10 registros

Filtrar datos: Excel PDF

	Proyecto	Actividad	Codigo Gasto	Factura	Recibo	Fecha	Proveedor
Ver	Proyecto ONG	P#1 A#2	g-00000000000014	3434		26/10/2015	Proveedor Electrico

Un total de 1 datos a mostrar (1 a 1)

← Pagina Anterior 1 Proxima Pagina →

- 4) Re direccionamientos a ejecuciones de la Actividades
 - a. Lista de Proyectos
 - b. Lista de Gastos de ActividadesRedirecciona a la lista según opción

Sistema de Control de Fondos y Donaciones

administrador

Gastos por Actividades-Proyectos

Proyecto: *** Seleccione ***

Actividad: *** Seleccione ***

Lista de Proyectos | Lista de Actividades | + Crear Nueva Gasto

Mostrar 10 registros

Filtrar datos: Excel PDF

	Proyecto	Actividad	Codigo Gasto	Factura	Recibo	Fecha	Proveedor
Ver	Proyecto ONG	P#1 A#2	g-00000000000014	3434		26/10/2015	Proveedor Electrico

Un total de 1 datos a mostrar (1 a 1)

← Pagina Anterior 1 Proxima Pagina →

5) Crear un nuevo Gasto-Actividad

Pulsar la opción de crear un nuevo gato, este re direccionara a otra página para registrar la información del gasto.

Sistema de Control de Fondos y Donaciones

administrador

Gastos por Actividades-Proyectos

Proyecto: *** Seleccione ***

Actividad: *** Seleccione ***

Lista de Proyectos | Lista de Actividades | **+ Crear Nueva Gasto**

Mostrar 10 registros

Filtrar datos: Excel PDF

	Proyecto	Actividad	Codigo Gasto	Factura	Recibo	Fecha	Proveedor
Ver	Proyecto ONG	P#1 A#2	g-00000000000014	3434		26/10/2015	Proveedor Electrico

Un total de 1 datos a mostrar (1 a 1)

← Pagina Anterior 1 Proxima Pagina →

Esta es la página para registrar la información correspondiente a un gasto:

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

Sistema de Control de Fondos y Donaciones

Proyecto: *** Seleccione *** (1)

Datos del Gasto

Actividad: *** Seleccione *** (2)

Proveedor: *** Seleccione *** (3)

Factura: (4) Recibo: (4)

Fecha: (5) IR: 0% (7)

TC US: (6) TC €: (8)

Detalle de Items

Entidad/ONG: *** Seleccione *** (9)

Item/Artículo: (10)

Cantidad: (11)

Costo Unitario: (12)

Descuento: 0% (13)

IVA: 0% (14)

+ Agregar Artículo/Item (15)

Entidad/ONG	Item/Artículo	Cantidad	Costo Unitario	IVA %	Descuento %	Total
No hay datos en la tabla						

Notas: (16)

Subtotal	0.00
IVA	0.00
Descuento	0.00
Total	0.00

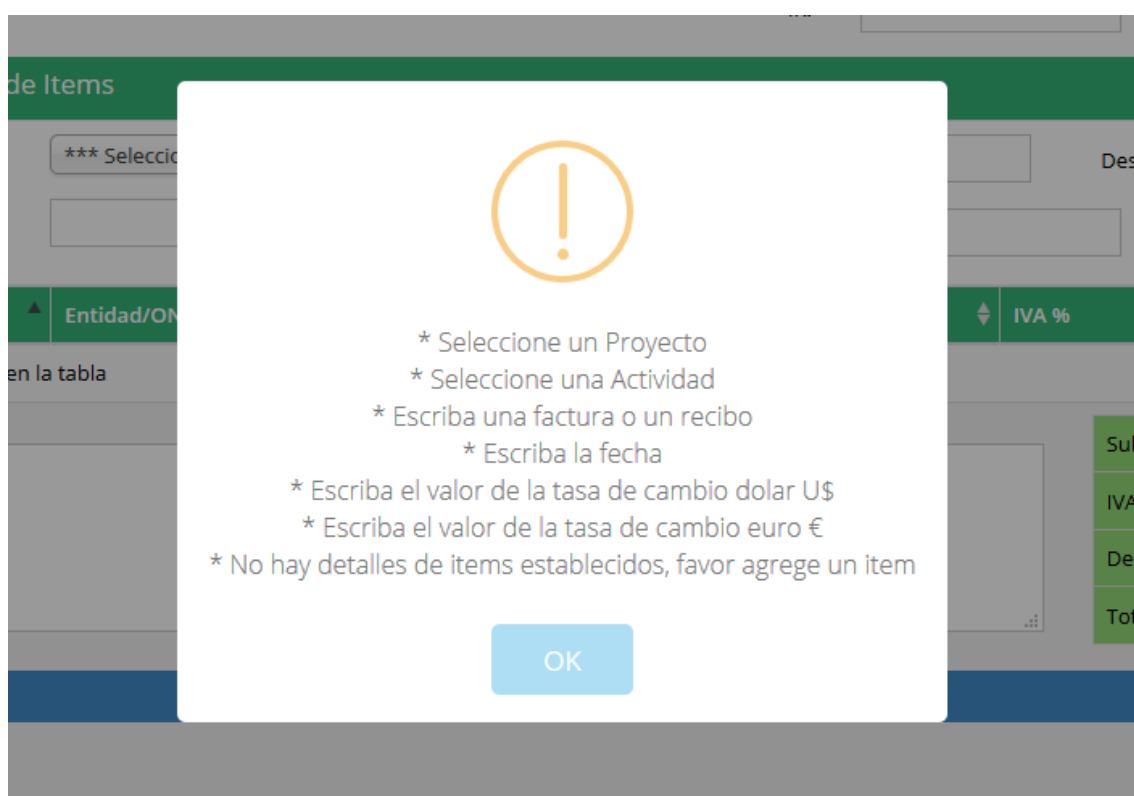
(17)

(18) Guardar Gasto

1. Proyecto del Gasto
2. Actividad según proyecto del Gasto
3. Proveedor
4. Factura/Recibo
5. Fecha del Gasto
6. Tasa de cambio dólar
7. IR
8. Tasa de Cambio Euros
9. Entidad según actividad que genera el gasto
10. Item o nombre del artículo
11. Cantidad del artículo
12. Costo unitario del Artículo o Item
13. Descuento del Item
14. IVA que se aplica por Item
15. Botón para agregar el ítem en la lista
16. Observaciones o nota a nivel general del gastos
17. Campos de visibilidad de la sumatoria total del gasto
18. Botón para guardar todos los registros o datos del Gasto en general.

A como en otros formularios, se validan los datos que son requeridos para su posterior almacenamiento o creación del gasto.

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA



6) Editar datos de un gasto seleccionado

Al pulsar la opción de editar, este redirección a la página como la de crear un gasto, con diferencia de que carga los datos correspondiente del gasto, listo para editar actualizar. Si las actividades finalizadas relacionadas a sus gastos, estos gastos no se presentara la opción de editar, de igual forma para borrar. A como también los gastos enlazadas a comprobantes contables no se permitirá la edición de los mismos.

- 7) Opción para borrar un gasto seleccionado
El gasto se borra, haciendo que el monto de la actividad aumente el saldo disponible que tenía en cuanto al proyecto.

2.6. Beneficiarios Donaciones

Este módulo compete al registro de las personas beneficiarios por las actividades realizadas en cada proyecto que se realizan, en resumen los beneficiarios. El siguiente grafico enumera las acciones y acceso del módulo:



1. Filtros de datos del grid (proyecto y actividades).
2. Filtro de escritura en el grid y opciones de exportación de los datos.
3. Crear un nuevo registro.

- 1) Filtros de datos del grid (proyecto y actividades).
- 2) Filtro de escritura en el grid y opciones de exportación de los datos.
- 3) Crear un nuevo registro

Al pulsar la opción de nuevo beneficiario, se abre un modal o ventana emergente para capturar la información y posteriormente guardarla.

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

1 Proyecto *** Seleccione ***

2 Actividad Select an Option

3 Beneficiario

4 Departamento

5 Municipio

6 Comarca

7 Descripción

Guardar Cancelar

- 1) Proyecto a cual se origina la actividad
- 2) Actividad a la cual se realizó la donación
- 3) Beneficiario (nombres y apellidos)
- 4) Departamento
- 5) Municipio
- 6) Comarca(si el caso lo amerita, para profundizar)
- 7) Descripción o observación

Proyecto Entregas de Zing Monimbo

Actividad Entrega de zing para los beneficiarios

Beneficiario Rosario del Carmen Rosales Duarte

Departamento Masaya

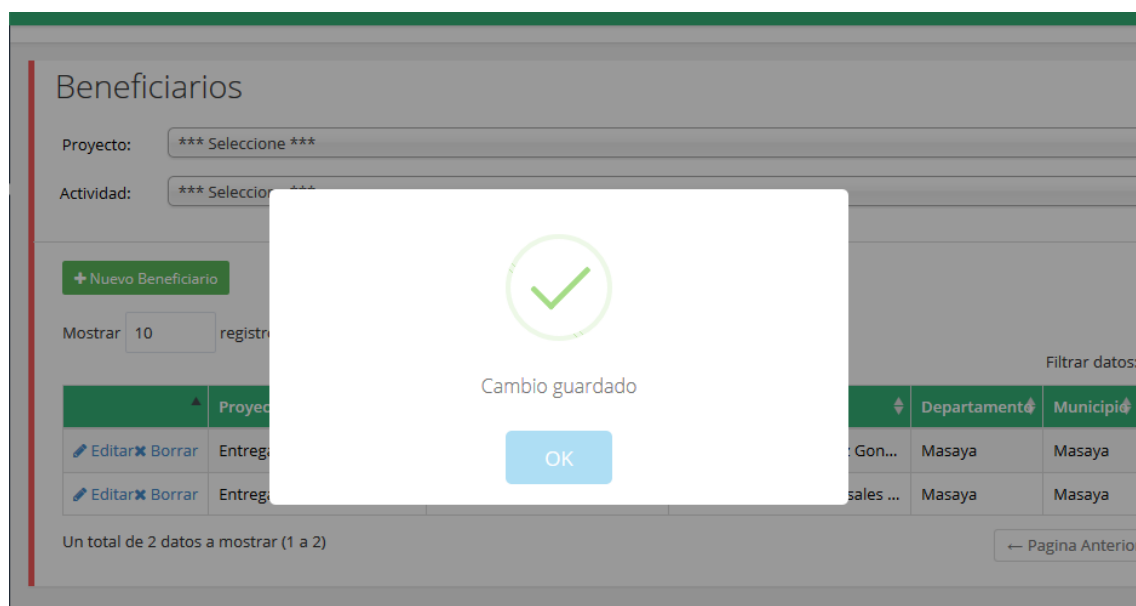
Municipio Masaya

Comarca Monimbo

Descripción

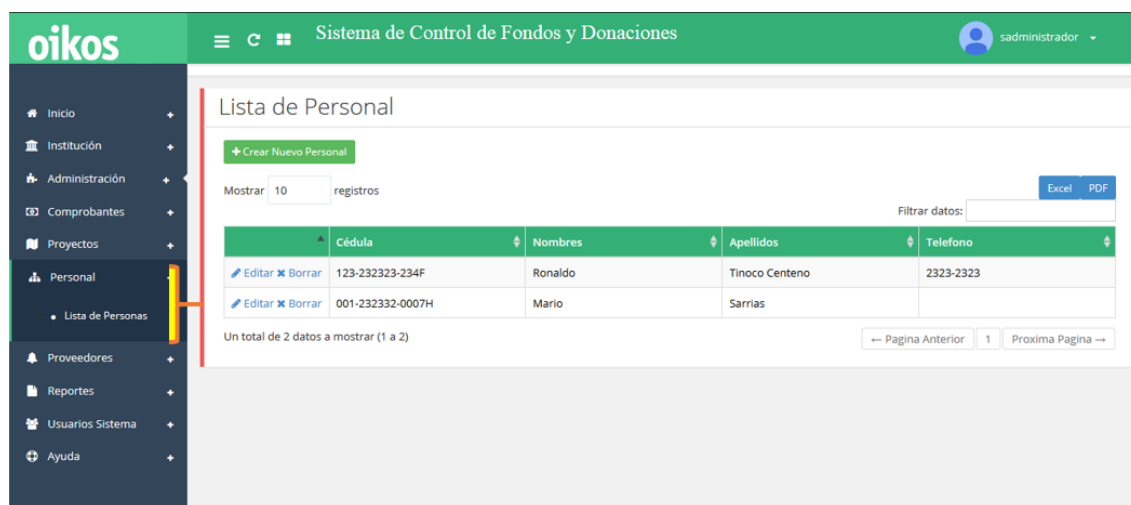
Guardar Cancelar

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA



3. Personal de las Entidades

Lista del Personal de las Entidades u ONG registradas en el transcurso de los proyectos, actividades, esta opción es para fin de poder actualizar la información almacenada para caso futuro. La funcionalidad es la misma como para los demás formularios, opciones de filtro, crear nuevos, editar y borrar.



1. Filtros de datos y opciones de exportación de datos.
2. Opción para crear o registrar un nuevo personal.
3. Opción para seleccionar un personal a editar o actualizar información.
4. Opción para Borrar un personal seleccionado.

- 1) Filtros de datos y opciones de exportación de datos.

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

Sistema de Control de Fondos y Donaciones

sadministrador

Lista de Personal

+ Crear Nuevo Personal

Mostrar 10 registros

Filtrar datos: Excel PDF

	Cédula	Nombres	Apellidos	Telefono
Editar Borrar	123-232323-234F	Ronaldo	Tinoco Centeno	2323-2323
Editar Borrar	001-232332-0007H	Mario	Sarrias	

Un total de 2 datos a mostrar (1 a 2)

← Pagina Anterior 1 Proxima Pagina →

- 2) Opción para crear o registrar un nuevo personal.
Al pulsar crear nuevo personal, se abre una ventana modal para el ingreso de la información de la persona.

Sistema de Control de Fondos y Donaciones

sadministrador

Lista de Personal

+ Crear Nuevo Personal

Mostrar 10 registros

Filtrar datos: Excel PDF

	Cédula	Nombres	Apellidos	Telefono
Editar Borrar	123-232323-234F	Ronaldo	Tinoco Centeno	2323-2323
Editar Borrar	001-232332-0007H	Mario	Sarrias	

Un total de 2 datos a mostrar (1 a 2)

← Pagina Anterior 1 Proxima Pagina →

Sistema de Control de Fondos y Donaciones

Personal

Nombres:

Apellidos:

Cédula: Activo: ☒

Dirección:

Teléfono:

Correo electrónico:

Cancelar Guardar

De igual forma que otros formularios, se valida los campos que son requeridos para posteriormente guardarlos.

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

- 3) Opción para seleccionar un personal a editar o actualizar información.

Esta opción es semejante a la registrar un nuevo registro de persona, se seleccón la persona y se abre la ventana modal con la información actual de la persona lista para su actualización de datos.



Sistema de Control de Fondos y Donaciones

administrador

Lista de Personal

+ Crear Nuevo Personal

Mostrar 10 registros

Filtrar datos:

Excel PDF

	Cédula	Nombres	Apellidos	Telefono
Editar Borrar	123-232323-234F	Ronaldo	Tinoco Centeno	2323-2323
Editar Borrar	001-232332-0007H	Mario	Sarrias	

Un total de 2 datos a mostrar (1 a 2)

← Pagina Anterior 1 Proxima Pagina →

- 4) Opción para Borrar un personal seleccionado.

Se validad si la persona no está relacionada a algún proyecto, en caso de ser así, se tendría que borrar los enlaces relacionados a la persona.



Sistema de Control de Fondos y Donaciones

administrador

Lista de Personal

+ Crear Nuevo Personal

Mostrar 10 registros

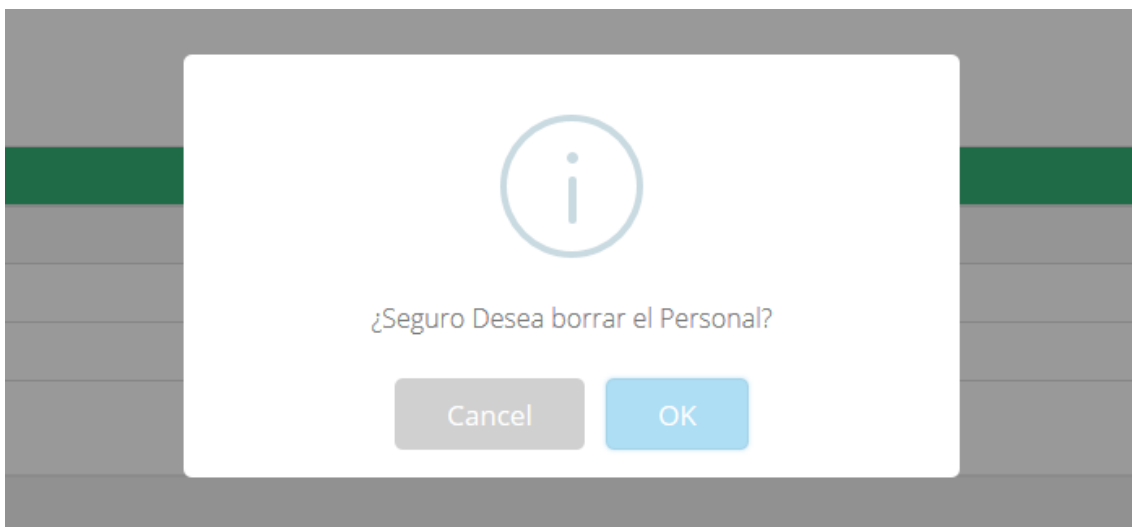
Filtrar datos:

Excel PDF

	Cédula	Nombres	Apellidos	Telefono
Editar Borrar	123-232323-234F	Ronaldo	Tinoco Centeno	2323-2323
Editar Borrar	001-232332-0007H	Mario	Sarrias	

Un total de 2 datos a mostrar (1 a 2)

← Pagina Anterior 1 Proxima Pagina →



¿Seguro Desea borrar el Personal?

Cancel OK

4. Proveedores

Lista de los Proveedores de la Entidad ONG, con el fin de llevar un catálogo de proveedores para los gastos en las actividades de los proyectos; para fin de poder actualizar la información registrada de los mismos si el caso lo amerita. La funcionalidad es la misma como para los demás formularios, opciones de filtro, crear nuevos, editar y borrar.

oikos Sistema de Control de Fondos y Donaciones administrador

Inicio
Institución
Administración
Comprobantes
Proyectos
Personal
Proveedores
Listado de Proveedores
Reportes
Usuarios Sistema
Ayuda

Lista de Proveedores

+ Nuevo Proveedor

Mostrar 10 registros

Filtrar datos: Excel PDF

	Proveedor	Dirección	Teléfono	Correo Electrónico
Editar Borrar	Aginsa	asdasdasda		
Editar Borrar	Proveedor Electrico			
Editar Borrar	Proveedor de certificado electrico2	adadasd	2323-2323	
Editar Borrar	dfsdfsdfsdf	dfsds		
Editar Borrar	Con saber	ertre		
Editar Borrar	yuiyuiyuiy	yuiyuiyuiy		
Editar Borrar	rterter	ertre		

Un total de 7 datos a mostrar (1 a 7)

← Pagina Anterior 1 Proxima Pagina →

1. Filtros de datos y opciones de exportación de datos.
2. Opción para crear o registrar un nuevo proveedor.
3. Opción para seleccionar un proveedor a editar o actualizar información.
4. Opción para Borrar un proveedor seleccionado.

1. Filtros de datos y opciones de exportación de datos.

Sistema de Control de Fondos y Donaciones administrador

Lista de Proveedores

+ Nuevo Proveedor

Mostrar 10 registros

Filtrar datos: Excel PDF

	Proveedor	Dirección	Teléfono	Correo Electrónico
Editar Borrar	Aginsa	asdasdasda		
Editar Borrar	Proveedor Electrico			
Editar Borrar	Proveedor de certificado electrico2	adadasd	2323-2323	
Editar Borrar	dfsdfsdfsdf	dfsds		
Editar Borrar	Con saber	ertre		
Editar Borrar	yuiyuiyuiy	yuiyuiyuiy		
Editar Borrar	rterter	ertre		

Un total de 7 datos a mostrar (1 a 7)

← Pagina Anterior 1 Proxima Pagina →

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

2. Opción para crear o registrar un nuevo proveedor.

Al pulsar la opción de nuevo proveedor, se abre una venta modal para el ingreso de la información a guardar.

The screenshot displays the 'Sistema de Control de Fondos y Donaciones' interface. At the top, there is a green header bar with the system name and a user profile icon labeled 'sadministrador'. Below the header, the main content area is titled 'Lista de Proveedores'. A red box highlights a green button labeled '+ Nuevo Proveedor'. Below this button, there is a 'Mostrar' dropdown set to '10' and a 'Filtrar datos:' input field. To the right of the filter field are 'Excel' and 'PDF' buttons. The main part of the interface is a table with the following columns: 'Proveedor', 'Dirección', 'Teléfono', and 'Correo Electrónico'. The table contains seven rows of data, each with an 'Editar' (pencil icon) and 'Borrar' (trash icon) link. The data rows are: 1. Aginsa, asdasdasdsa, empty, empty; 2. Proveedor Electrico, empty, empty, empty; 3. Proveedor de certificado electrico2, adadasd, 2323-2323, empty; 4. dfsdfsdfsdf, dsfsds, empty, empty; 5. Con saber, ertre, empty, empty; 6. yuiyuiyuiy, yuiyuiyuiy, empty, empty; 7. rterter, ertterte, empty, empty. Below the table, it says 'Un total de 7 datos a mostrar (1 a 7)'. At the bottom right of the table area are navigation buttons: '← Pagina Anterior', '1', and 'Proxima Pagina →'. Overlaid on the bottom half of the screenshot is a modal form titled 'Proveedor'. The form has four input fields: 'Nombre Proveedor:', 'Dirección:', 'Teléfono:', and 'Correo electrónico:'. At the bottom of the modal are two buttons: 'Cancelar' and 'Guardar'.

Proveedor	Dirección	Teléfono	Correo Electrónico
Aginsa	asdasdasdsa		
Proveedor Electrico			
Proveedor de certificado electrico2	adadasd	2323-2323	
dfsdfsdfsdf	dsfsds		
Con saber	ertre		
yuiyuiyuiy	yuiyuiyuiy		
rterter	ertterte		

De igual forma que otros formularios, se valida los campos que son requeridos para posteriormente guardarlos.

3. Opción para seleccionar un proveedor a editar o actualizar información.

Esta opción es semejante a la registrar un nuevo registro de proveedor, se seleccón el proveedor y se abre la ventana modal con la información actual del proveedor lista para su actualización de datos.

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

Sistema de Control de Fondos y Donaciones

sadministrador

Lista de Proveedores

+ Nuevo Proveedor

Mostrar 10 registros

Excel PDF

Filtrar datos:

	Proveedor	Dirección	Teléfono	Correo Electrónico
Editar Borrar	Aginsa	asdasdasdsa		
Editar Borrar	Proveedor Electrico			
Editar Borrar	Proveedor de certificado electrico2	adadasd	2323-2323	
Editar Borrar	dfsdfsdfsdf	dsfsds		
Editar Borrar	Con saber	ertre		
Editar Borrar	yuliyuluy	yuliyuluy		
Editar Borrar	rterter	ertertre		

Un total de 7 datos a mostrar (1 a 7)

← Pagina Anterior

1

Proxima Pagina →

4. Opción para Borrar un proveedor seleccionado.

Se valida si el proveedor no está relacionada a algún gasto, en caso de ser así, se tendría que borrar los enlaces relacionados al proveedor.

Sistema de Control de Fondos y Donaciones

sadministrador

Lista de Proveedores

+ Nuevo Proveedor

Mostrar 10 registros

Excel PDF

Filtrar datos:

	Proveedor	Dirección	Teléfono	Correo Electrónico
Editar Borrar	Aginsa	asdasdasdsa		
Editar Borrar	Proveedor Electrico			
Editar Borrar	Proveedor de certificado electrico2	adadasd	2323-2323	
Editar Borrar	dfsdfsdfsdf	dsfsds		
Editar Borrar	Con saber	ertre		
Editar Borrar	yuliyuluy	yuliyuluy		
Editar Borrar	rterter	ertertre		

Un total de 7 datos a mostrar (1 a 7)

← Pagina Anterior

1

Proxima Pagina →



5. Comprobantes Contables

Los comprobantes contables, que en si son generados de los recibos y/o facturas generados en las actividades que son programadas en los proyectos. Estos depende únicamente que el recibo y/o factura estén ingresados en el sistema, sin importar el estado del proyecto y sus actividades.

A continuación presentamos las opciones y ejecución para el módulo de comprobantes contables del sistema:



MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

1. Buscar las facturas (se abre un modal para la selección)
2. Datos de factura o recibo seleccionado
3. Dato del proveedor del recibo o factura seleccionado.
4. Dato de fecha del recibo o factura seleccionado
5. Dato del monto total del gasto realizado, (sirve para validar el total del comprobante).
6. Numero de Comprobante actual.
7. Fecha del comprobante.
8. Opción para agregar unos ajustes contables en caso especial del comprobante.
9. Guardar los datos del Comprobante.

- 1) Buscar las facturas (se abre un modal para la selección)
Buscar el gasto a registrar en un comprobante contable, se abre una opción modal para la selección.



- 2) Datos de factura o recibo seleccionado
- 3) Dato del proveedor del recibo o factura seleccionado.
- 4) Dato de fecha del recibo o factura seleccionado
- 5) Dato del monto total del gasto realizado, (sirve para validar el total del comprobante).
- 6) Numero de Comprobante actual.
- 7) Fecha del comprobante.

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

Comprobante de Diario

Recibo/Factura: 23000 Proveedor: Hotelito Kelly

Comprobante: 000078 Fecha: 09/02/2016 Fecha Gasto: 26/01/2016 Monto Total: 1,220.00

Item	Item Total	Cuenta Contable	Debe	Haber
hospedaje	1,220.00	*** Seleccione la Cuenta Contable ***		

Observaciones del comprobante

Debe: 0.00 Haber: 0.00

8) Opción para agregar unos ajustes contables en caso especial del comprobante.

Agregar un ajustes para igualar el debe y haber del comprobante.

Agregar Ajuste

Monto o Valor: 1,220.00

Se añade al grid automáticamente con el monto escrito.

Comprobante de Diario

Recibo/Factura: 23000 Proveedor: Hotelito Kelly

Comprobante: 000078 Fecha: 09/02/2016 Fecha Gasto: 26/01/2016 Monto Total: 1,220.00

Item	Item Total	Cuenta Contable	Debe	Haber
Borrar	1,220.00	*** Seleccione la Cuenta Contable ***		
hospedaje	1,220.00	*** Seleccione la Cuenta Contable ***		

Observaciones del comprobante

Debe: 0.00 Haber: 0.00

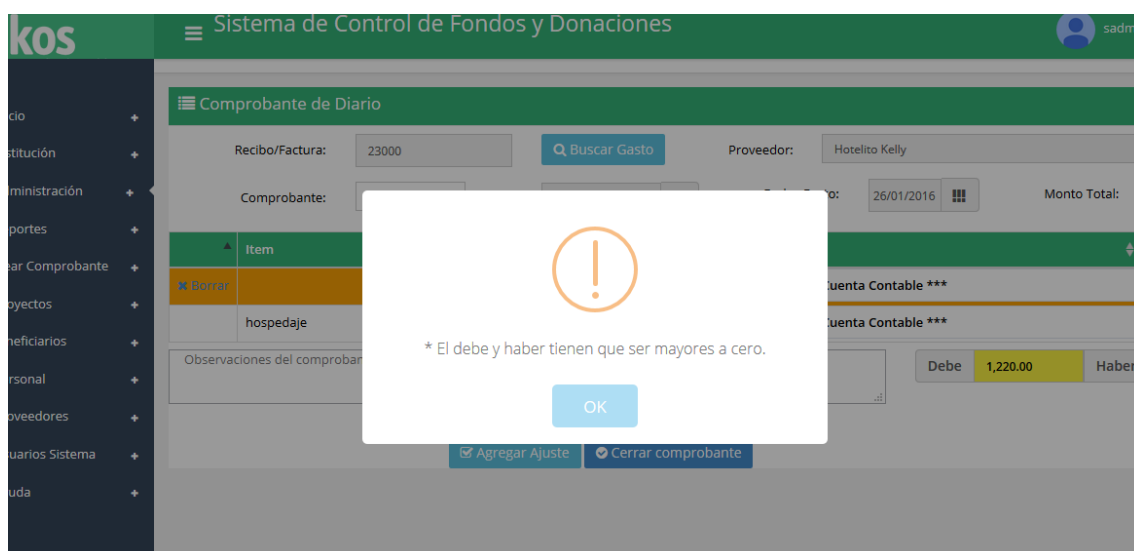
*** Seleccione la Cuenta Contable ***

- 100 Capital social
- 101 Fondo social
- 102 Capital
- 200 Gastos de investigación
- 213 Maquinaria
- 400 Proveedores
- 401 Proveedores, efectos comerciales a pagar
- 404 Proveedores, empresas asociadas
- 405 Proveedores, otras partes vinculadas
- 600 Compras y gastos
- 601 Compras a Proveedores
- 602 Compras externas de Holcim
- 300 Existencias
- 301 Materias primas
- 302 Otros aprovisionamientos

9) Guardar los datos del Comprobante.

En estas últimas opciones existe validación de campos requeridos. (Selección de cuenta contables.)

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA



Una vez ingresados los datos necesarios para procesar el comprobante contable son guardados.

Item	Item Total	Cuenta Contable	Debe	Haber
Borrar	1,220.00	320 Elementos y conjuntos incorporables	<input type="checkbox"/>	<input checked="" type="checkbox"/>
hospedaje	1,220.00	600 Compras y gastos	<input checked="" type="checkbox"/>	<input type="checkbox"/>

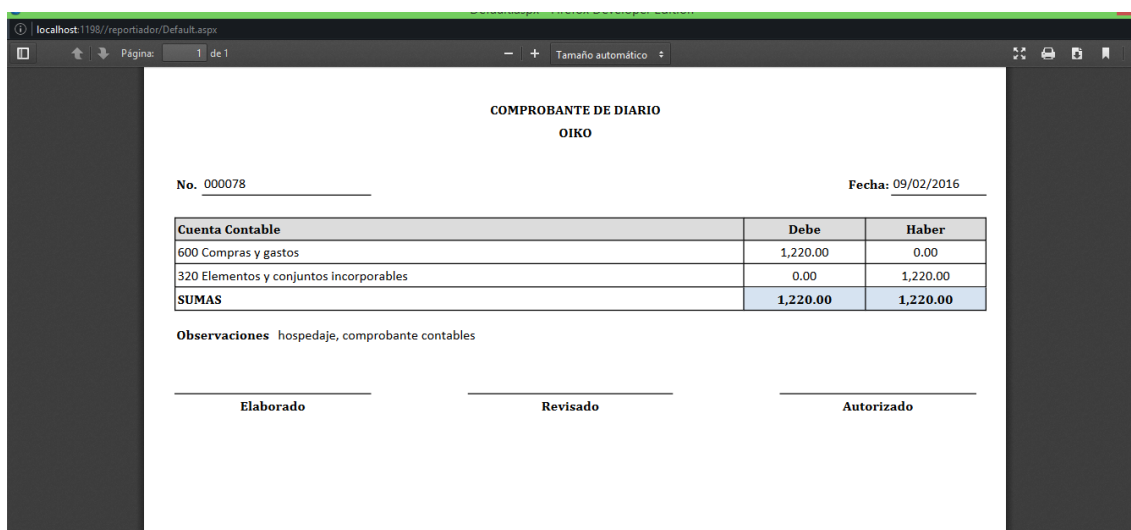
hospedaje, comprobante contables

Debe 1,220.00 Haber 1,220.00

[Agregar Ajuste](#) [Cerrar comprobante](#)

Los valores correspondientes al debe y haber se reflejan de color resaltado amarillo para su mejor visualización, ya que estos tienen que ser de igual valor.

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA



6. Gestión de Usuarios del Sistema

La gestión de usuarios en el sistema es una parte importante para poder gestionar permisos y accesos al sistema, el sistema permite estos procesos para su uso y accesibilidad. Esta opción es únicamente visible para usuarios administradores.

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA



1. Filtros de datos y opciones de exportación de datos.
2. Opción para crear o registrar un nuevo usuario.
3. Opción para seleccionar un usuario a editar o actualizar información.
4. Cambiar Contraseña de un Usuario.

1) Filtros de datos y opciones de exportación de datos.



- 2) Opción para crear o registrar un nuevo usuario.
Al pulsar la opción de nuevo usuarios, re direcciona a otra página para registrar los datos.

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

Sistema de Control de Fondos y Donaciones

administrador

Lista de Usuarios del Sistema

+ Agregar un nuevo Usuario

Mostrar 10 registros

Excel PDF

Filtrar datos:

	Usuario	Nombre(s)	Apellido(s)	Email	Roles
Editar Contraseña	operador1	operador1	operador1	opeador1@yahoo.es	Operador
Editar Contraseña	user1	user1	user1	jmayorga92@hotmail.com	Consulta

Un total de 2 datos a mostrar (1 a 2)

← Pagina Anterior 1 Proxima Pagina →

oikos

Sistema de Control de Fondos y Donaciones

administrador

Usuario del Sistema

Inicio de Sesión:

Nombre(s) de Usuario:

Apellido(s) de Usuario:

Contraseña:

Confirmar Contraseña:

Roles Asignados:

Correo Electrónico:

Los campos necesarios son validados evitando que puedan ser guardados.

3) Opción para seleccionar un usuario a editar o actualizar información.

Al editar se re direccionar a una página con la información registrada del usuario, a excepción del nombre de usuario que no se puede cambiar.

Sistema de Control de Fondos y Donaciones

administrador

Lista de Usuarios del Sistema

+ Agregar un nuevo Usuario

Mostrar 10 registros

Excel PDF

Filtrar datos:

	Usuario	Nombre(s)	Apellido(s)	Email	Roles
Editar Contraseña	operador1	operador1	operador1	opeador1@yahoo.es	Operador
Editar Contraseña	user1	user1	user1	jmayorga92@hotmail.com	Consulta

Un total de 2 datos a mostrar (1 a 2)

← Pagina Anterior 1 Proxima Pagina →

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

 Sistema de Control de Fondos y Donaciones

 Usuario del Sistema

Inicio de Sesión:

operador1

Nombre(s) de Usuario:

operador1

Apellido(s) de Usuario:

operador1

Roles Asignados:

Operador x

Correo Electrónico:

opeador1@yahoo.es





Cancelar

Guardar


4) Cambiar Contraseña de un Usuario.

Para los Usuarios administradores, podrá realizar cambio de contraseña para los demás usuarios, caso en que algún usuario olvide su contraseña o requiera algún cambio posterior por alguna razón.

Al pulsar cambio de contraseña, este te re direcciona a otra página con un formulario para cambiar la contraseña:

 Sistema de Control de Fondos y Donaciones  administrador





Lista de Usuarios del Sistema

 Agregar un nuevo Usuario

Mostrar 10 registros

Excel PDF

Filtrar datos:

Usuario	Nombre(s)	Apellido(s)	Email	Roles
  operador1	operador1	operador1	opeador1@yahoo.es	Operador
  user1	user1	user1	jmayorga92@hotmail.com	Consulta




Un total de 2 datos a mostrar (1 a 2)


← Pagina Anterior

1

Proxima Pagina →

MANUAL DE USUARIO DE SISTEMA DE CONTROL Y FONDOS Y DONACIONES DE PROYECTOS DE OIKOS DE NICARAGUA

 Sistema de Control de Fondos y Donaciones

 Cambio de Contraseña de Usuario

Inicio de Sesión:

operador1

Nombre(s) de Usuario:

operador1

Apellido(s) de Usuario:

operador1

Nueva Contraseña:

Confirmar Contraseña Nueva:

Cancelar

Guardar